

SON-2243

PATENT APPLICATION

#5

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In the Patent Application of  
Kuniaki TAKAHASHI et al

Group Art Unit: To Be Assigned

Serial No. To Be Assigned

Examiner: To Be Assigned

Filed: October 11, 2001

For: MOTION VECTOR CONVERSION METHOD AND  
CONVERSION APPARATUS

CLAIM TO PRIORITY UNDER 35 U.S.C. 119

Commissioner for Patents  
Washington, D.C. 20231

J1046 U.S. PTO  
09/973749  
  
10/11/01

Sir:

The benefit of the filing date of the following prior application filed in the following foreign country is hereby requested and the right of priority provided under 35 U.S.C. 119 is hereby claimed:

Japanese Patent Appl. No. P2000-312309 filed October 12, 2000

In support of this claim, filed herewith is a certified copy of said original foreign application.

Respectfully submitted,

Ronald P. Kananen  
Reg. No. 24,104

Dated: October 11, 2001

RADER, FISHMAN & GRAUER P.L.L.C.  
1233 20<sup>TH</sup> Street, NW, Suite 501  
Washington, DC 20036  
202-955-3750-Phone  
202-955-3751 - Fax  
Customer No. 23353

50/P1-181800

J1046 U.S. PTO  
09/973749  
10/11/01

日本国特許庁  
JAPAN PATENT OFFICE

別紙添付の書類に記載されている事項は下記の出願書類に記載されて  
いる事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed  
with this Office

出願年月日  
Date of Application:

2000年10月12日

出願番号  
Application Number:

特願2000-312309

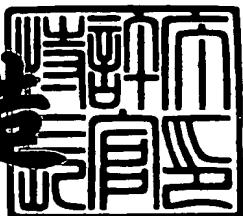
出願人  
Applicant(s):

ソニー株式会社

2001年 8月31日

特許庁長官  
Commissioner,  
Japan Patent Office

及川耕造



出証番号 出証特2001-3080803

【書類名】 特許願  
【整理番号】 0000729206  
【提出日】 平成12年10月12日  
【あて先】 特許庁長官殿  
【国際特許分類】 H03M 7/30  
【発明者】  
【住所又は居所】 東京都品川区北品川6丁目7番35号 ソニー株式会社  
内  
【氏名】 高橋 邦明  
【発明者】  
【住所又は居所】 東京都品川区北品川6丁目7番35号 ソニー株式会社  
内  
【氏名】 佐藤 数史  
【発明者】  
【住所又は居所】 東京都品川区北品川6丁目7番35号 ソニー株式会社  
内  
【氏名】 鈴木 輝彦  
【特許出願人】  
【識別番号】 000002185  
【氏名又は名称】 ソニー株式会社  
【代表者】 出井 伸之  
【代理人】  
【識別番号】 100080883  
【弁理士】  
【氏名又は名称】 松隈 秀盛  
【電話番号】 03-3343-5821  
【手数料の表示】  
【予納台帳番号】 012645  
【納付金額】 21,000円

【提出物件の目録】

【物件名】 明細書 1

【物件名】 図面 1

【物件名】 要約書 1

【包括委任状番号】 9707386

【ブルーフの要否】 要

【書類名】 明細書

【発明の名称】 動きベクトル変換方法及び変換装置

【特許請求の範囲】

【請求項1】 飛び越し走査のMPEG2の画像圧縮情報を示すビットストリームを入力とし、順次走査のMPEG4の画像圧縮情報を示すビットストリームを出力とする画像情報変換方法における動きベクトル変換方法において、

上記入力となるMPEG2の画像圧縮情報を示すビットストリームにおけるMPEG2の $16 \times 16$ の動きベクトルを入力とし、MPEG4の $8 \times 8$ の動きベクトル及びMPEG4の $16 \times 16$ の動きベクトルを生成し、MPEG2のビットストリームのPフレームを一枚置きに破棄することにより、フレームレートを下げた、低ビットレートのMPEG4のビットストリームを生成する動きベクトル変換方法であって、

動きベクトル補整を兼ね、入力となるMPEG2の画像圧縮情報を示すビットストリーム内のマクロブロック及びブロック毎の情報を予め格納しておき、

該格納された情報を基に、破棄する直前のPフレームの動きベクトルを複製し、該複製された動きベクトルを時間方向に2倍に伸張して、MPEG2のイントラフレームより変換されたMPEG4の動きベクトルを生成することを特徴とする動きベクトル変換方法。

【請求項2】 飛び越し走査のMPEG2の画像圧縮情報を示すビットストリームを入力とし、順次走査のMPEG4の画像圧縮情報を示すビットストリームを出力とする画像情報変換方法における動きベクトル変換方法において、

上記入力となるMPEG2の画像圧縮情報を示すビットストリームにおけるMPEG2の $16 \times 16$ の動きベクトルを入力とし、MPEG4の $8 \times 8$ の動きベクトル及びMPEG4の $16 \times 16$ の動きベクトルを生成し、MPEG2のビットストリームのIフレーム又はPフレームを一枚置きに破棄することにより、フレームレートを下げた、低ビットレートのMPEG4のビットストリームを生成する動きベクトル変換方法であって、

MPEG2のイントラフレームより変換されたMPEG4の動きベクトルの符号化効率を判定するパラメータとして、入力となる直前に変換されたMPEG2

の画像圧縮情報を示すビットストリームにおけるマクロブロック及びブロック毎の動きベクトル情報を利用して、動きベクトル〇の予測との比較を行い、予測残差の少ない動きベクトルを符号化効率の高いものとして選択することを特徴とする動きベクトル変換方法。

【請求項3】 飛び越し走査のMPEG2の画像圧縮情報を示すビットストリームを入力とし、順次走査のMPEG4の画像圧縮情報を示すビットストリームを出力とする画像情報変換方法における動きベクトル変換方法において、

上記入力となるMPEG2の画像圧縮情報を示すビットストリームにおけるMPEG2の $16 \times 16$ の動きベクトルを入力とし、MPEG4の $8 \times 8$ の動きベクトル及びMPEG4の $16 \times 16$ の動きベクトルを生成し、MPEG2のビットストリームのIフレーム又はPフレームを一枚置きに破棄することにより、フレームレートを下げた、低ビットレートのMPEG4のビットストリームを生成する動きベクトル変換方法であって、

動きベクトル補整を兼ね、入力となるMPEG2の画像圧縮情報を示すビットストリーム内の動きベクトル情報を基に、MPEG2の動きベクトルより動きベクトル変換方法により変換されたMPEG4の $8 \times 8$ の動きベクトルを入力とし、該入力動きベクトル値を中心とした動きベクトルの再検索を行って、動きベクトルを補整することにより、一つのマクロブロックを構成する4つの $8 \times 8$ の動きベクトルのうち、最も予測残差の少ないMPEG4の $8 \times 8$ の動きベクトルを $16 \times 16$ の動きベクトルに割り当てることにより、 $16 \times 16$ の動きベクトルを生成することを特徴とする動きベクトル変換方法。

【請求項4】 飛び越し走査のMPEG2の画像圧縮情報を示すビットストリームを入力とし、順次走査のMPEG4の画像圧縮情報を示すビットストリームを出力とする画像情報変換装置における動きベクトル変換装置において、

上記入力となるMPEG2の画像圧縮情報を示すビットストリームにおけるMPEG2の $16 \times 16$ の動きベクトルを入力とし、MPEG4の $8 \times 8$ の動きベクトル及びMPEG4の $16 \times 16$ の動きベクトルを生成する動きベクトル変換装置であって、

MPEG2のビットストリームのPフレームを一枚置きに破棄する破棄手段を

設け、該破棄手段によって、フレームレートを下げた、低ビットレートのMPEG4のビットストリームを生成すると共に、

動きベクトル補整装置を兼ね備え、入力となるMPEG2の画像圧縮情報を示すビットストリーム内のマクロブロック及びブロック毎の情報を予め格納しておく格納手段と、

該格納手段に格納された情報を基に、破棄する直前のPフレームの動きベクトルを複製し、該複製された動きベクトルを時間方向に2倍に伸張して、MPEG2のイントラフレームより変換されたMPEG4の動きベクトルを生成する動きベクトル生成手段とを有することを特徴とする動きベクトル変換装置。

【請求項5】 飛び越し走査のMPEG2の画像圧縮情報を示すビットストリームを入力とし、順次走査のMPEG4の画像圧縮情報を示すビットストリームを出力とする画像情報変換装置における動きベクトル変換装置において、

上記入力となるMPEG2の画像圧縮情報を示すビットストリームにおけるMPEG2の $16 \times 16$ の動きベクトルを入力とし、MPEG4の $8 \times 8$ の動きベクトル及びMPEG4の $16 \times 16$ の動きベクトルを生成する動きベクトル変換装置であって、

MPEG2のビットストリームのIフレーム又はPフレームを一枚置きに破棄する破棄手段を設け、該破棄手段によって、フレームレートを下げた、低ビットレートのMPEG4のビットストリームを生成すると共に、

MPEG2のイントラフレームより変換されたMPEG4の動きベクトルの符号化効率を判定するパラメータとして、入力となる直前に変換されたMPEG2の画像圧縮情報を示すビットストリームにおけるマクロブロック及びブロック毎の動きベクトル情報を利用して、動きベクトル0の予測との比較を行い、予測残差の少ない動きベクトルを符号化効率の高いものとして選択する選択手段を有することを特徴とする動きベクトル変換装置。

【請求項6】 飛び越し走査のMPEG2の画像圧縮情報を示すビットストリームを入力とし、順次走査のMPEG4の画像圧縮情報を示すビットストリームを出力とする画像情報変換装置における動きベクトル変換装置において、

上記入力となるMPEG2の画像圧縮情報を示すビットストリームにおけるM

MPEG2の $16 \times 16$ の動きベクトルを入力とし、MPEG4の $8 \times 8$ の動きベクトル及びMPEG4の $16 \times 16$ の動きベクトルを生成する動きベクトル変換装置であって、

MPEG2のビットストリームのIフレーム又はPフレームを一枚置きに破棄する破棄手段を設け、該破棄手段によって、フレームレートを下げた、低ビットレートのMPEG4のビットストリームを生成すると共に、

動きベクトル補整装置を兼ね備え、入力となるMPEG2の画像圧縮情報を示すビットストリーム内の動きベクトル情報を基に、MPEG2の動きベクトルより動きベクトル変換装置により変換されたMPEG4の $8 \times 8$ の動きベクトルを入力とし、該入力動きベクトル値を中心とした動きベクトルの再検索を行って、動きベクトルを補整することにより、一つのマクロブロックを構成する4つの $8 \times 8$ の動きベクトルのうち、最も予測残差の少ないMPEG4の $8 \times 8$ の動きベクトルを $16 \times 16$ の動きベクトルに割り当てるにより、 $16 \times 16$ の動きベクトルを生成する動きベクトル生成手段を有することを特徴とする動きベクトル変換装置。

#### 【発明の詳細な説明】

##### 【0001】

##### 【発明の属する技術分野】

本発明は、離散コサイン変換等の直交変換と動き補償によって圧縮された画像情報を示すビットストリームを、衛星放送、ケーブルTV、インターネットなどのネットワークメディアを介して受信する際に、若しくは光、磁気ディスク、フラッシュメモリのような記憶メディア上で処理する際に用いられる、MPEG方式を利用した画像情報変換方法（画像情報変換装置）における動きベクトル変換方法（ベクトル変換装置）に関するものである。

##### 【0002】

##### 【従来の技術】

近年、画像情報をデジタルデータとして取り扱い、その際、効率の高い情報の伝送、蓄積を目的とし、画像情報特有の冗長性を利用して、離散コサイン変換等の直交変換と動き補償により圧縮するMPEG方式に準拠した装置が、放送局な

どの情報配信、及び一般家庭における情報受信の双方において普及しつつある。

#### 【0003】

特に、MPEG2（ISO/IEC 13818-2）は、汎用画像符号化方式として定義されており、飛び越し走査画像及び順次走査画像の双方、並びに標準解像度画像及び高精細画像を網羅する標準方式で、プロフェッショナル用途及びコンシューマー用途の広範なアプリケーションに今後とも用いられるものと予想される。MPEG2の圧縮方式を用いることにより、例えば $720 \times 480$ 画素を持つ標準解像度の飛び越し走査画像や、 $4 \sim 8 \text{ Mbps}$ 、 $1920 \times 1080$ 画素を持つ高解像度の飛び越し走査画像であれば、 $18 \sim 22 \text{ Mbps}$ の符号量（ビットレート）を割り当てることで、高い圧縮率と良好な画質の実現が可能である。

#### 【0004】

MPEG2は主として放送用に適合する高画質符号化を対象としていたが、MPEG1より低い符号量（ビットレート）、つまりより高い圧縮率の符号化方式には対応していなかった。携帯端末の普及により、今後そのような符号化方式のニーズは高まると思われる。これに対応して、MPEG4符号化方式の標準化が行われた。MPEG4の画像符号化方式に関しては、1998年12月にISO/IEC 14496-2としてその規格が国際標準として承認された。

#### 【0005】

ところで、デジタル放送用に一度符号化されたMPEG2の画像圧縮情報を示すビットストリームを、携帯端末上等で処理するのにより適した、より低い符号量（ビットレート）のMPEG4の画像圧縮情報を示すビットストリームに変換したいというニーズがある。

#### 【0006】

かかる目的を達成する画像情報変換装置の従来例を図1に示し、以下これについて説明する。この図1の画像情報変換装置は、入力されたMPEG2の画像圧縮情報を示すビットストリームを、MPEG4の画像圧縮情報を示すビットストリームへ変換する装置である。すなわち、入力されたMPEG2の画像圧縮情報を示すビットストリームは、MPEG2の画像復号化装置1により復元される。

復元された画像信号は、解像度・フレームレート変換装置2へ伝送され、任意の異なる解像度及びフレームレートを持つ画像信号に変換される。変換された画像信号は、MPEG4の画像符号化装置3に入力され、MPEG4の画像情報符号化装置3によりMPEG4の画像圧縮情報を示すビットストリームに符号化されて出力される。

#### 【0007】

従来の画像情報変換装置では図1に示したように、MPEG2の復号化方式により復元された画像信号を、MPEG4の画像符号化装置により符号化し、MPEG4の画像圧縮情報を示すビットストリームを出力する。MPEG2の画像情報復号化装置1においては、水平及び垂直方向成分ともに、入力となるMPEG2の画像圧縮情報を示すビットストリームの8次のDCT(離散コサイン変換)係数すべてを用いた復号処理を行うことが考えられる。しかし、垂直方向には8次の係数すべてを用いるもの、水平方向には8次係数のうち低域4成分のみを用いた復号処理(以下これを $4 \times 8$ ダウンデコードと呼ぶ)、若しくは、水平、垂直方向ともに水平方向には8次係数のうち低域4成分のみを用いた復号処理(以下これを $4 \times 4$ ダウンデコードと呼ぶ)を行うことで、画質劣化を最小限に抑えながら、演算量とビデオメモリ容量を削減し、更に後段のダウンサンプリング処理を簡略化する構成も考えられる。

#### 【0008】

このような従来方法では、MPEG4の画像符号化装置において、入力された画像信号を符号化する際、動きベクトルを検出する演算処理量は、全演算処理量の約60~70パーセントを占める。そのため、画像のリアルタイムでの処理が困難となり、時間遅延が発生する、装置が大規模になってしまふ等の問題点を生じる。

#### 【0009】

かかる問題を解決する手段として、筆者らは先に、図2に示した画像情報変換装置(本願出願時未公知)(先行例)を提案した。

#### 【0010】

ところで、図3は、MPEG2の画像圧縮情報を示すビットストリーム中の動

きベクトルと、MPEG4の画像圧縮情報を示すビットストリーム中の動きベクトルの相関を示す図である。図3A、Bは、解像度変換前及び後の現フレームの画像をそれぞれ示す。画像の解像度が変換される際、変換後の前フレームでの位置から現フレームでの位置への動きベクトルの水平成分は、解像度変換前の動きベクトルの水平成分と、画像の横方向の解像度変換レートによって求めることができる。解像度変換後の垂直成分は、解像度変換前の動きベクトルの垂直成分と、画像の縦方向の解像度変換レートによって求められる。すなわち、解像度変換前の動きベクトルと、変換後の動きベクトルと大きな相関を持つ。その相関を利用し、解像度変換前の動きベクトルから変換後の動きベクトルを求めることができる。

## 【0011】

すなわち、図2の画像情報変換装置では、入力されたMPEG2の画像圧縮情報を示すビットストリームを、MPEG2のマクロブロックの動きベクトルやマクロブロックタイプ等のパラメータを利用して、MPEG4の動きベクトルへ簡潔に変換する。MPEG4の画像情報符号化装置7内では、動きベクトルの検出を行わず、変換された動きベクトルを用いた画像符号化を行う。結果として、MPEG4の画像符号化装置内での動き検出を行わないため、処理量が大幅に減らされる。

## 【0012】

このようにMPEG2の動きベクトルから、MPEG4の動きベクトルへの変換を行うこと、また動きベクトル以外にも、MPEG2での復号に用いたパラメータ、若しくは変換した後のパラメータを採用することにより、MPEG4の画像情報符号化装置7の処理量が減らされるので、この装置7による信号の時間遅延を少なくすることができる。

## 【0013】

図3において、入力されたMPEG2の画像圧縮情報を示すビットストリームは、図2のMPEG2の画像情報復号化装置4において、復号化処理が施されて、MPEG4の画像圧縮情報を示すビットストリームが出力される。MPEG2の画像情報復号化装置4においては、水平及び垂直方向成分とともに、入力となる

MPEG2の画像圧縮情報を示すビットストリームの8次のDCT係数すべてを用いた復号処理を行うことが考えられるが、 $4 \times 8$  ダウンデコード、若しくは $4 \times 4$  ダウンデコードを行うことで、画質劣化を最小限に抑えながら、演算量とビデオメモリ容量を削減し、更に後段のダウンサンプリング処理を簡略化する構成も考えられる。復号化装置4より出力された画像信号が、解像度フレームレート変換装置5へ送られ、解像度・フレームレート変換を行った後、外部から入力された画像サイズ調整フラグにより、MPEG4の画像符号化に適した解像度を持つ画像信号が出力される。

#### 【0014】

ここで、解像度フレームレート変換装置5では、まず、MPEG2の画像復号化装置4より入力された画像信号を、解像度フレーム変換器によって解像度変換を行う。ここでは、縦・横とも解像度を $1/2$ にする場合を例に挙げる。図4に示しているように、縦方向のダウンサンプリングは、入力された飛び越し走査画像の第一フィールド、若しくは第二フィールドを抽出し、順次走査画像に変換する。横方向は、ダウンサンプリングフィルタを用いて、 $1/2$ の解像度に変換する。また、低ビットレートを実現するために、解像度の変換による圧縮だけではなく、I/Pピクチャのみ第一フィールド若しくは第二フィールドを抜き出し、時間方向においてフレームレートを落とす。たとえば、図4に示したMPEG2のIBBPの画像は解像度・フレーム変換後、IPの第一フィールドの構成になる。解像度・フレーム変換を行った画像は、MPEG4の画像符号化方式により符号化できるように、縦、横の画素数がともに16の倍数にする。そのため、外部から入力される画像サイズ調整フラグにより、画素の補填あるいは画素の除去回路によって、画素の補填あるいは画素の除去を行う。

#### 【0015】

画像サイズ調整フラグは、解像度フレームレート変換装置5の外部から入力され、画像の縦・横の画素数が16の倍数でない場合に対し、画像への画素補填若しくは、除去を判別するためのフラグである。

#### 【0016】

図5を参照して、画像サイズ調整フラグによる画像への処理を説明する。MP

E G 2 の画像復号化装置 4 より出力された画像の解像度が  $m$  画素 ×  $n$  画素であるとすると、 $m$ 、 $n$  は共に 16 の倍数であるが、縦、横ともに  $1/2$  にダウンサンプリングされた  $m/2$ 、 $n/2$  は、16 の整数倍、若しくは、16 で割って 8 画素の余りがある。 $m/2$ 、 $n/2$  が共に 16 の倍数の場合においては、M P E G 4 の符号化方式に適する画像になっているため、画像への処理は行わない。それ以外は、M P E G 4 符号化方式に適しないため、画像サイズ調整フラグによる画像への処理が必要となる。画像サイズ調整フラグには、画素の補填と除去の二つの選択肢を持つ。 $m/2$  若しくは  $n/2$  を 16 で割って、8 画素が余った場合においては、画像の除去を選択すれば、余った 8 画素を除去する。すなわち、出力画像は  $(m/2 - 8)$  若しくは  $(n/2 - 8)$  となる。一方画素の補填を選択すれば、新たに作成した 8 画素若しくは元の画像から複製した 8 画素若しくは画像に適した 8 画素を行い、あるいは列の先頭若しくは、後部から付け加える。

すなわち、出力画像は  $(m/2 + 8)$  若しくは  $(n/2 + 8)$  となる。結果として、変換後の画像解像度の横と縦は 16 の倍数となり、M P E G 4 の符号化方式に適したサイズを持つ画像が出力される。

#### 【0017】

一方、入力されたM P E G 2 の画像圧縮情報を示すビットストリームは、M P E G 2 の画像復号化装置 4 では、可変長符号の復号化を行った後、P ピクチャのみのマクロブロック動きベクトルやマクロブロックタイプなどその他のパラメータが取り出され、動きベクトル変換装置 6 へ伝送される。

#### 【0018】

動きベクトル変換装置 6 における動きベクトル変換方法を、図 6 を用いて説明する。図 6 A および、図 6 B の実線で区切られている各正方格子の一つ一つがマクロブロックを示している。図 6 A はM P E G 2 の復号化装置 4 より出力された画像、すなわち、解像度変換前の画像である。図 6 B は図 6 A の画像を解像度フレームレート変換装置 5 により、縦・横の解像度とも  $1/2$  に変換された画像である。たとえば、変換前の図 6 A の左上の斜線の付された  $16 \times 16$  マクロブロックは、図 6 B の左上の斜線の付された  $8 \times 8$  ブロックに変換される。すなわち、図 6 A に示した 4 つの網掛けされた  $16 \times 16$  マクロブロックが、解像度変換

された後、それぞれ図6Bに示した4つの網掛けされた $8 \times 8$ ブロックに変換され、それらによって1つの $16 \times 16$ マクロブロックが構成される。解像度変換前と後の動きベクトルの相関が大きいため、変換後の $8 \times 8$ ブロックの動きベクトルは、変換前の $16 \times 16$ マクロブロックの動きベクトルより求めることができる。さらに、4つの $8 \times 8$ の動きベクトルから1つの $16 \times 16$ の動きベクトルを求め、MPEG4の画像符号化方式により符号化するのに用いる動きベクトルの4つの $8 \times 8$ の動きベクトルと、1つの $16 \times 16$ の動きベクトルが生成される。

#### 【0019】

図2の画像情報変換装置における動きベクトル変換装置6の詳細ブロック図を示す図7を参照して、動きベクトル変換の動作原理を説明する。入力されたMPEG2の画像圧縮情報を示すビットストリーム中の動きベクトル、画像サイズなどのパラメータは、MPEG2の $16 \times 16$ の動きベクトル→MPEG4の $8 \times 8$ の動きベクトル変換装置8によって、補整前の $8 \times 8$ の動きベクトルが作成される。

#### 【0020】

図8のフローチャートを参照して、このMPEG2の $16 \times 16$ の動きベクトル→MPEG4の $8 \times 8$ の動きベクトル変換装置8の動作原理を説明する。入力されたMPEG2のマクロブロックの動きベクトルとマクロブロックタイプに対し、MPEG2の $16 \times 16$ の動きベクトル→MPEG4の $8 \times 8$ の動きベクトル変換装置8は次のように動作する。飛び越し走査のMPEG2の画像圧縮情報を示すビットストリームにおいては、一般的にフレーム構造が用いられるため、今回はフレーム構造の場合への処理のみの変換方法を説明する。

#### 【0021】

ステップST-1で、入力されたMPEG2のマクロブロックの動きベクトルとマクロブロックタイプに対し、イントラマクロブロック、スキップマクロブロック及びインターマクロブロックのいずれであるかの判別を行う。イントラマクロブロックである場合には、MPEG2のイントラマクロブロックは、解像度変換後の $8 \times 8$ ブロックが動きベクトルを持つことを想定する。その処理としては

、ステップST-2で、まず $8 \times 8$ の動きベクトルを0に設定し、さらに、補整器11による処理を行うため、イントラモードフラグを設ける。MPEG2では、イントラマクロブロックである場合、イントラモードフラグが立つ。

## 【0022】

スキップマクロブロックである場合には、ステップST-3で、各ブロックの動きベクトルを0と設定する。

## 【0023】

図9を参照して、インターマクロブロックである場合における、画像がフレーム構造、フレーム予測である場合の動きベクトル変換の概念を説明する。図9Aは解像度変換前を、図9Bは解像度変換後をそれぞれ示す。図3で説明したように、変換後の動きベクトルの水平成分は変換前の動きベクトルの水平成分と画像の横方向の解像度変換レートから求められる。垂直成分は変換前の動きベクトルの垂直成分と画像の縦方向の解像度変換レートから求められる。すなわち、横方向の解像度を $1/2$ に変換した際、変換後の動きベクトルの水平成分も変換前の $1/2$ になる。縦方向の解像度を $1/2$ に変換した際、変換後の動きベクトルの垂直成分も変換前の $1/2$ になる。

## 【0024】

図9に示した動きベクトルは、たとえば、変換前の(8, 12)から変換後の(4, 6)になる。尚、この場合は、整数画素の中間値(半画素)の間隔を1としている。図には、図9Aの解像度変換前では、黒丸は整数画素の位置を示し、菱形は半画素の位置を示している。図9Bの解像度変換後では、半画素を白丸で示す。図からわかるように、変換前整数画素の位置に示している動きベクトルは、変換後整数画素若しくは半画素の位置に示すが、変換前半画素の位置に示している動きベクトルは解像度変換後参照する画素がなくなる。そこで、変換前の動きベクトルが半画素の位置を示している場合、変換後の動きベクトルも予測画像の半画素の位置を示すようとする。それは本来、復号化された画像信号には、量子化による歪みが含まれているため、そのまま予測画像として使用すると予測効率が低下し、画質劣化を引き起こす場合がある。これを低減するために、低域通過フィルタに相当する参照画面での各画素間を1:1で直線補間した半画素精度

を選択される場合もあり、これにより、画質劣化を避けることができる。従って、MPEG4の画像符号化方式による符号化も、予測効率を向上し、画質劣化を防ぐために、MPEG4で動きベクトルが半画素の位置に示している場合、MPEG4のフォーマットに変換した際も半画素の位置に示すように変換する。変換前と変換後の動きベクトルの対応関係を図10の表図に示す。

## 【0025】

次に、画像がフレーム構造でフィールド予測である場合だが、まず、第一フィールド予測である時について、図11に動きベクトル変換の概念図を示す。動きベクトルの水平成分は、図10と同様な処理を行われる。垂直方向は、第一フィールドを抽出することにより、解像度を1/2に変換することを可能とする。また、予測も第一フィールド予測を行っているため、変換前の動きベクトルは、そのまま変換後の動きベクトルになる。

## 【0026】

第二フィールド予測である場合の動きベクトル変換の概念図を図12及び図13に示す。解像度変換した際、第一フィールドのみが抽出されるため、変換後は第一フィールドの画素値を参照画像として用いる。そのため、MPEG2で予測画像として用いた第二フィールドの画素値を、解像度変換後の第一フィールド予測に変換するように、動きベクトルの時空間補整を行う。図12は、第二フィールド予測から近似的に第一フィールド予測に変換するための空間的な補整を行う手法を示した図である。図12Aは解像度変換前を、図12Bは解像度変換後をそれぞれ示す。すなわち、動きベクトルの垂直成分に1を加える。図からわかるように、第二フィールド予測で求めた動きベクトルの垂直成分に1を足すと、1行を繰り上げられることによって、第二フィールドが第一フィールドと同様の空間位置に達し、空間上で、第一フィールド予測で求めた動きベクトルのようになる。数1の式は空間補整により、第一フィールドと同様な空間位置にある第二フィールド、すなわち近似第一フィールドを予測とした時、動きベクトルMV<sub>top</sub>の垂直成分を表している。

## 【0027】

## 【数1】

垂直成分：近似 $MV_{top} = MV_{bottom} + 1$

【0028】

また、飛び越し走査のMPEG2の画像圧縮情報には第一フィールドと第二フィールドとで時間ずれがある。そのため、第二フィールドから近似された第一フィールドと実際の第一フィールドとの時間ずれの補整を行う。図13は各フィールドの時間的位置関係を示している。ここで、第一フィールドと第二フィールドの間隔を1とし、aをIピクチャの第二フィールドとPピクチャの第一フィールドの間隔とすると、aは1, 3, 5, 7, ……のような奇数になる。尚、aが1の場合は、画像の構成がIPP………の場合である。時間補整した動きベクトルMV'を、数2の式に示す。

【0029】

【数2】

垂直成分： $MV' = \{ (a+1) / a \}$  近似 $MV_{top}$

【0030】

数1の式を数2の式に代入すると、変換後の動きベクトルの垂直成分は数3の式のようになる。

【0031】

【数3】

垂直成分： $MV' = \{ (a+1) / a \} (MV_{bottom} + 1)$

【0032】

尚、変換後の動きベクトルの水平成分は、変換前の動きベクトルに $(a+1) / a$ を乗じ、時間的な補整を行った後、図10の表図に示す計算に従って求められる。

【0033】

動きベクトルの垂直成分に対して、場合により、時間的補整を行った後、空間的な補整を行ってもよい。その場合、動きベクトルMV'の垂直成分を、数4の式に示す。尚、水平成分は空間・時間補整（空間補整が行ってから時間補整を行う）と時間・空間補整（時間補整が行ってから空間補整を行う）は同様な値となる。

【0034】

【数4】

$$\text{垂直成分: } MV' = \{ (a + 1) / a \} MV_{\text{bottom}} + 1$$

【0035】

数3の式と数4の式の差、すなわち、空間・時間補整を行った場合と、時間・空間補整を行った場合の動きベクトルの垂直成分の差は  $1/a$  になる。従って、 $a$  の値によって、その差による影響が異なるので、 $a$  が 1 の場合と、1 より大きい、すなわち 3, 5, 7, …… の 2 つの場合における補整方法を説明する。

【0036】

まず、 $a = 1$  の場合について、数3の式の  $a$  に 1 を代入すると、動きベクトルの垂直成分は数5の式のようになる。

【0037】

【数5】

$$\text{垂直成分: } MV' = 2 \times (MV_{\text{bottom}} + 1)$$

【0038】

数4の式の  $a$  に 1 を代入すると、動きベクトルの垂直成分は数6の式のようになる。

【0039】

【数6】

$$\text{垂直成分: } MV' = 2 \times (MV_{\text{bottom}} + 1) - 1$$

【0040】

その結果、変換前の動きベクトル  $MV_{\text{bottom}}$  に 0, 1, 2, …… を代入すると、数5の式による値は 2, 4, 6, …… のような偶数になる。すなわち、空間・時間補整を行うと、変換前の動きベクトルは整数画素の位置に示しても、半画素の位置に示しても、変換後はすべて整数画素の位置に来る。また、数6の式による値は、1, 3, 5, …… のような奇数になる。すなわち、時間・空間補整を行うと、変換前の動きベクトルは整数画素の位置に示しても半画素の位置に示しても、変換後はすべて半画素の位置に来る。従って、変

換前に整数画素の位置に示している動きベクトルは、変換後も整数画素の位置に来るようとする場合、空間・時間補整を行う。また、変換前の半画素の位置に示している動きベクトルは、変換後も半画素の位置に示すようとする場合、時間・空間補整を行う。すなわち、変換前の動きベクトルに対し、空間補整及び時間補整を交互に使用し、解像度変換後の動きベクトルに変換するか、若しくは、変換前の動きベクトルに対して、すべてが時間・空間補整を行う。

#### 【0041】

以上の動きベクトル変換処理が終了した後、補整前のMPEG4の $8 \times 8$ の動きベクトルが出力される。出力された $8 \times 8$ の動きベクトルは、画像サイズ調整フラグによる動きベクトル調整器9(図7)に伝送され、外部より入力された画像サイズ調整フラグにより、画像サイズに適した動きベクトルを出力する。

#### 【0042】

画像サイズ調整フラグによる動きベクトル調整器9における動作を、図14のフローチャートを参照して説明する。入力画像サイズm画素×n画素に対して、ステップST-11で、 $m/2$ ,  $n/2$ が共に16の倍数であるか否かを判別し、 $m/2$ ,  $n/2$ が共に16の倍数であれば、動きベクトル変換装置6から出力されたMPEG4の $8 \times 8$ の動きベクトルを、処理することなくそのまま出力する。 $m/2$ ,  $n/2$ のいずれかが16の倍数でない場合、外部から入力された画像サイズ調整フラグが作動し、ステップST-12で画素除去か否かが判別され、画素除去の場合は、除去された8画素の $8 \times 8$ の動きベクトルを出力しないで、他の $8 \times 8$ の動きベクトルを出力する。画素除去でない場合は、ステップST-13で、画素補填か否かの判別を行い、画素補填の場合は、補填された8画素の $8 \times 8$ の動きベクトルを0に設定し、他の入力された $8 \times 8$ の動きベクトルと合わせて出力される。

#### 【0043】

再び図7に戻って説明する。画像サイズ調整フラグによる動きベクトル調整器9より出力された画像サイズに適した $8 \times 8$ の動きベクトルは、MPEG4の $8 \times 8$ の動きベクトル→MPEG4の $16 \times 16$ の動きベクトル変換装置10、あるいは図16のMPEG4の $8 \times 8$ の動きベクトル→MPEG4の $16 \times 16$ の

動きベクトル変換装置12によって変換される。

## 【0044】

図7の動きベクトル変換装置10においては、マクロブロックを構成する4つのブロックの中、イントラでないマクロブロックから変換されたブロックの動きベクトルの和をイントラでないマクロブロックから変換されたブロックの数で割った平均を $16 \times 16$ の動きベクトルとして出力する。

## 【0045】

図16に戻って説明する。第一の方法として、MPEG4の $8 \times 8$ の動きベクトル→MPEG4の $16 \times 16$ の動きベクトル変換装置15においては、MPEG2の $16 \times 16$ の動きベクトル→MPEG4の $8 \times 8$ の動きベクトル変換装置12において生成された、当該マクロブロックに対するMPEG4の $8 \times 8$ の動きベクトルのうち、もっとも符号化効率が高いと考えられるマクロブロックより生成されたものを選択し、MPEG4の $16 \times 16$ の動きベクトルとして出力する。

## 【0046】

符号化効率の判定は、マクロブロック情報バッファ14に格納された、画像情報変換装置の入力となるMPEG2の画像圧縮情報を示すビットストリームにおけるマクロブロック毎の情報を基に行う。

## 【0047】

すなわち、第一の方法は、4つのマクロブロックのうち、最も非零DCT係数の少ないマクロブロックを符号化効率が高いとする方法である。第二の方法は、4つのマクロブロックのうち、輝度成分のDCT係数に割り当てられたビット数の最も少ないマクロブロックを符号化効率が高いとする方法である。第三の方法は、4つのマクロブロックのうち、DCT係数に割り当てられたビット数の最も少ないマクロブロックを符号化効率が高いとする方法である。第四の方法は、4つのマクロブロックのうち、動きベクトル等を含めた、マクロブロックに割り当てられた全ビット数の最も少ないマクロブロックを符号化効率が高いとする方法である。第五の方法は、4つのマクロブロックのうち、割り当てられた量子化スケールが最も小さいマクロブロックを符号化効率が高いとする方法である。第六

の方法は、4つのマクロブロックのうち、コンプレクシティの最も低いマクロブロックを符号化効率が高いとする方法である。各マクロブロックに割り当てられたコンプレクシティXは、そのマクロブロックに割り当てられた量子化スケールQ、及びビット数Bを用いて、数7の式のように計算される。

【0048】

【数7】

$$X = Q \cdot B$$

【0049】

ここで、Bは、マクロブロック全体に割り当てられたビット数でも良いし、DCT係数に割り当てられたビット数でも良いし、輝度成分に割り当てられたDCT係数に割り当てられたビット数でも良い。

【0050】

再び図16に戻って説明する。第二の方法として、MPEG4の $8 \times 8$ の動きベクトル→MPEG4の $16 \times 16$ の動きベクトル変換装置15においては、MPEG2の $16 \times 16$ の動きベクトル→MPEG4の $8 \times 8$ の動きベクトル変換装置12において生成された、当該マクロブロックに対するMPEG4の $8 \times 8$ の動きベクトルのうち、もっとも重み付けが高いと考えられるマクロブロックより生成されたものを2回重複して数え、合計5つの $8 \times 8$ の動きベクトルのうち、中間値の長さを持つ動きベクトルをもっとも符号化効率が高いと選択し、MPEG4の $16 \times 16$ の動きベクトルとして出力する。ここで動きベクトルの長さの比較は水平方向、垂直方向のそれぞれの長さの二乗根の和を用い、平方根を求める処理は省略する。

【0051】

重み付けの判定は、マクロブロック情報バッファ14に格納された、画像情報変換装置の入力となるMPEG2の画像圧縮情報を示すビットストリームにおけるマクロブロック毎の情報を元に行う。

【0052】

すなわち、第一の方法は、4つのマクロブロックのうち、最も非零DCT係数の少ないマクロブロックを重み付けが高いとする方法である。第二の方法は、4

つのマクロブロックのうち、輝度成分のDCT係数に割り当てられたビット数の最も少ないマクロブロックを重み付けが高いとする方法である。第三の方法は、4つのマクロブロックのうち、DCT係数に割り当てられたビット数の最も少ないマクロブロックを重み付けが高いとする方法である。第四の方法は、4つのマクロブロックのうち、動きベクトル等を含めた、マクロブロックに割り当てられた全ビット数の最も少ないマクロブロックを重み付けが高いとする方法である。第五の方法は、4つのマクロブロックのうち、割り当てられた量子化スケールが最も小さいマクロブロックを重み付けが高いとする方法である。第六の方法は、4つのマクロブロックのうち、コンプレクシティの最も低いマクロブロックを重み付けが高いとする方法である。各マクロブロックに割り当てられたコンプレクシティXは、そのマクロブロックに割り当てられた量子化スケールQ、及びビット数Bを用いて以下のように計算される。

【0053】

【数8】

$$X = Q \cdot B$$

【0054】

ここで、Bは、マクロブロック全体に割り当てられたビット数でも良いし、DCT係数に割り当てられたビット数でも良いし、輝度成分に割り当てられたDCT係数に割り当てられたビット数でも良い。

【0055】

再び図7に戻って説明する。画像サイズ調整フラグによる動きベクトル調整器9より出力された画像サイズに適した8×8の動きベクトルは、MPEG2のイントラマクロブロックに対する動きベクトル補整器11に入力される。そこでMPEG2の画像圧縮情報を示すビットストリームにおいてはイントラであったマクロブロックから変換されたブロックの8×8の動きベクトルは、MPEG4の8×8の動きベクトル→MPEG4の16×16の動きベクトル変換装置10によって求められた16×16の動きベクトルに置きかえることによって補整される。補整後の8×8の動きベクトルと、MPEG4の8×8の動きベクトル→MPEG4の16×16の動きベクトル変換装置10で求めた16×16の動きベ

クトルを合わせて、MPEG4動きベクトルとして出力する。

#### 【0056】

図7のMPEG2のイントラマクロブロックに対する動きベクトル補整器11の詳細構成図を示す図15を参照して、その動きベクトル補整の動作原理を説明する。画像サイズに適した $8 \times 8$ の動きベクトルがMPEG2のイントラマクロブロックに対する補整器11に入力し、イントラモードフラグが立っている場合、すなわち、MPEG2の画像圧縮情報を示すビットストリームにおいてはイントラマクロブロックである場合、スイッチSWの可動接点mが固定接点a側に倒れ、置換装置PKで、そのイントラマクロブロックから変換された $8 \times 8$ ブロックの動きベクトルを、MPEG4の $8 \times 8$ の動きベクトル→MPEG4の $16 \times 16$ の動きベクトル変換装置10より求められた $16 \times 16$ の動きベクトルに置き換えられる。この場合、そのイントラマクロブロックの周辺に存在するインターマクロブロックの動きベクトルから変換された動きベクトルに置換しても良く、若しくは、イントラマクロブロックに一番近いインターマクロブロックの動きベクトルから変換された動きベクトルに置換してもよい。尚、4つのブロックともイントラマクロブロックから変換された場合は、4つとも動きベクトルが0となり、MPEG4の $8 \times 8$ の動きベクトル→MPEG4の $16 \times 16$ の動きベクトル変換装置10によって求められる $16 \times 16$ の動きベクトルも0となるため、MPEG4の画像符号化に用いる動きベクトルは0となり、マクロブロックタイプがイントラモードとなる。また、イントラモードフラグが立たない場合、すなわち、MPEG2ではイントラマクロブロックがない場合、スイッチSWの可動接点mが固定接点b側に倒れ、入力された $8 \times 8$ の動きベクトルがそのまま出力される。

#### 【0057】

次に生成された動きベクトルの再検索を行う場合について説明する。インターマクロブロックに対し、画像がフレーム構造、フレーム予測である場合の動きベクトル変換の概念図を図17に示す。図17Aは解像度変換前を、図17Bは解像度変換後をそれぞれ示す。図3で説明したように、変換後の動きベクトルの水平成分は変換前の動きベクトルの水平成分と画像の横方向の解像度変換レートか

ら求められる。垂直成分は変換前の動きベクトルの垂直成分と画像の縦方向の解像度変換レートから求められる。すなわち、横方向の解像度を $1/2$ に変換した際、変換後の動きベクトルの水平成分も変換前の $1/2$ になる。縦方向の解像度を $1/2$ に変換した際、変換後の動きベクトルの垂直成分も変換前の $1/2$ になる。図9に示した動きベクトルは、たとえば、変換前の(8, 12)から変換後の(4, 6)になる。尚、この場合は、整数画素の中間値(半画素)の間隔を1としている。図17Aの解像度変換前では、黒丸は整数画素の位置を示し、菱形は半画素の位置を示している。図17Bの解像度変換後では、半画素は白丸で示す。

#### 【0058】

図からわかるように、変換前の整数画素の位置に示している動きベクトルは、変換後整数画素若しくは半画素の位置に示すが、変換前半画素の位置に示している動きベクトルは解像度変換後、参照する画素がなくなる。そこで、変換前の動きベクトルが半画素の位置を示している場合、変換後の動きベクトルも予測画像の最も近い整数画素位置を示すようとする。これは、後に動きベクトル補整において動きベクトル周辺画素を再検索する際に、サーチウィンドウの中心画素値を整数画素にするためである。まず、サーチウィンドウ内で最も予測誤差が少ない整数画素の動きベクトルを求め、次にその整数画素周辺の9の半画素値を検索することにより処理のステップを削減している。変換前と変換後の動きベクトルの対応関係を図19に示す。

#### 【0059】

以上、これに限らず、図9、10、11、12に示されたように半画素動きベクトルを生成し、これを中心に再検索を行うことも可能であるが、この場合半画素値をサーチウィンドウ内で事前に求める行程が必要となり、処理工数が増えることとなる。

#### 【0060】

次に、画像がフレーム構造でフィールド予測である場合だが、まず、第一フィールド予測である時について、図19の動きベクトル変換の概念図を参照して説明する。図19Aは解像度変換前を、図19Bは解像度変換後をそれぞれ示す。

動きベクトルの水平成分は、図18で説明したのと同様な処理を行われる。垂直方向は、第一フィールドを抽出することにより、解像度を1/2に変換することを可能とする。また、予測も第一フィールド予測を行っているため、変換前の動きベクトルは、そのまま変換後の動きベクトルになる。

**【0061】** 第二フィールド予測である場合の動きベクトル変換の概念図を図20に示す。図20Aは解像度変換前を、図20Bは解像度変換後をそれぞれ示す。解像度変換した際、第一フィールドのみが抽出されるため、変換後は第一フィールドの画素値を参照画像として用いる。フィールドのライン補整及び時間方向の動きベクトル補整は図12で説明している方法と同様に補整を行い、その後水平方向の画素値は変換後に整数画素を示すように、図18と同様な処理が行われる。

### **【0062】**

次に図21を用いて再検索による動きベクトル補整処理を説明する。すなわち、動きベクトル補整装置は、図2の動きベクトル変換装置6により、スケーリングされた動きベクトルを入力とし、符号化効率を高めるために再検索により補整された動きベクトルを出力とする。また、動きベクトル補整方向情報フラグによるサーチウィンドウのサイズを設定する。動きベクトル補整方向情報フラグに関しては後に図23及び24を用いて説明する。

### **【0063】**

すなわち、動きベクトル補整装置はより精度が高い動きベクトルを求めるために動きベクトル変換装置6により動きベクトルのスケーリングにより生じた歪みを再検索によって補整する。まず、入力されたMPEG4の8×8の動きベクトルの周辺画素を水平方向±2整数画素、垂直方向±1画素の動きベクトル検索を行う。これにより、動きベクトル変換装置6による求められた動きベクトルを中心とするサーチウィンドウを水平5画素、垂直3画素に抑え、動きベクトル検索の処理数を大幅に減少させることが可能となる。以上、サーチウィンドウを水平5画素、垂直3画素に設定したが、ここではこれに限らず、水平、垂直方向のサーチウィンドウの検索画素数は任意に選択しても良い。また、検索処理数を削減させるために後に説する動きベクトル補整方向情報フラグを用いて、動きベクトルの

補整用サーチウインドウの大きさを動きベクトルの進行方向に対し、順方向、逆方向非対称に設定しても良い。

#### 【0064】

図22に動きベクトル変換装置（本願出願時に未公知の先行例）の構成ブロック図を示す。動きベクトル変換装置は図2の動きベクトル変換装置6に示したものと同様である。まず、変換装置221では、上述した動きベクトルの空間・時間補整を行って、 $8 \times 8$ の動きベクトルを生成する。次に、検索装置222において、 $8 \times 8$ の動きベクトルに上述したように、 $8 \times 8$ の動きベクトルの再検索処理を行う。例えば、生成された $8 \times 8$ の動きベクトルの参照先を中心に、検索サーチウインドウを整数画素で縦横2画素ずつで検索を行う。これにより $8 \times 8$ の動きベクトルの予測精度を向上させることが可能である。次に一つのマクロブロックを構成する4つの $8 \times 8$ の動きベクトルそれぞれに対して再検索された予測誤差及び、再検索処理を施した $8 \times 8$ の動きベクトルを、変換装置223へ入力する。

#### 【0065】

変換装置223では4つの $8 \times 8$ の動きベクトルのうち、 $8 \times 8$ の動きベクトル再検索時に求められた予測残差のうち、最も予測誤差が少ない動きベクトルを判別し、 $16 \times 16$ の動きベクトルに割り当てる。次に生成された $16 \times 16$ の動きベクトルを上述と同様に検索装置214において再検索処理を行って、 $16 \times 16$ の動きベクトルの予測精度を向上させる。これによりMPEG4の $8 \times 8$ の動きベクトル及び、 $16 \times 16$ の動きベクトルを生成し、出力する。

#### 【0066】

出力された $8 \times 8$ の動きベクトル及び $16 \times 16$ の動きベクトルは、それぞれ半画素検索装置225に入力され、半画素精度で $8 \times 8$ 及び $16 \times 16$ の動きベクトルが出力される。

#### 【0067】

ここで図23及び図24を用いて図21及び図22に示した動きベクトル補整方向情報フラグについて説明する。動きベクトル補整装置は動きベクトル補整方向情報フラグにより動きベクトルの進行方向に対して、再検索サーチウインドウ

のサイズを設定する。まず、図23Aに示すように変換前のMPEG2動きベクトルが整数画素であり変換後のMPEG4動きベクトルも整数画素値である場合は、同位相の画素が存在し、動きベクトルの画素値に歪みが生じないため、動きベクトル補整のサーチウィンドウは動きベクトルの進行方向に対して対称に設定する。図23Bに示すように変換前のMPEG2動きベクトルが整数画素値であり、変換後のMPEG4動きベクトルが半画素精度である場合、動きベクトルの進行方向に対し、繰り上げにより近似のMPEG4の整数画素値に修正するため、サーチウィンドウは動きベクトルの進行方向に対し、逆方向に大きくとることにより検索処理数を削減することが可能となる。例えば、動きベクトルと逆方向に2整数画素値、順方向に①整数画素値で検索する。勿論、動きベクトル変換時にMPEG4の動きベクトルを繰り下げにより近似のMPEG4の整数画素値とした場合、サーチウィンドウの設定は動きベクトルの進行方向に対して順方向に大きく設定する。

#### 【0068】

図24A、Bの場合も同様に変換前のMPEG2動きベクトルが半画素精度を示し、MPEG4動きベクトルに変換する際に最も近いMPEG4の整数画素値に修正するため、図23Bのケースと同様に歪みが生じる方向を基にMPEG4の動きベクトル補整装置のサーチウィンドウを動きベクトルの進行方向に対し、順方向または逆方向に大きく設定することが出来る。これにより、動きベクトル再検索の処理数を減少させるため、動きベクトルの歪みの方向を動きベクトル装置6より出力し、動きベクトル補整装置に入力し、動きベクトルの補整検索を行うのに、最適なサーチウィンドウの大きさを設定することが出来る。

#### 【0069】

MPEG4の画像符号化装置7では、解像度フレームレート変換装置5からの出力画像を受け取り、動きベクトル変換装置6から出力されたMPEG4の動きベクトルを用いて、MPEG4の画像符号化方式により符号化を行い、MPEG4の画像圧縮情報を示すビットストリームを出力する。

#### 【0070】

以下に、本発明者らが発明した動きベクトル変換方法及び変換装置（本願出願

時未公知)について説明する。

## 【0071】

図25にMPEG2ストリームからMPEG4ストリームへのピクチャタイプとVOP(Video Object Plane)タイプの対応概念図を示す。GOP(画面グループ)構造がN15M3である毎秒30フレームのMPEG2のストリームをGOP構造N5M1である毎秒5フレームのMPEG4のストリームに変換する場合、I及びPフレーム(I-frame, P-frame)を一枚置きにMPEG4に変換する。図25に示すようMPEG2のストリームのN15M1の2GOPにMPEG4のストリームのN5M1の1GOVが対応するため、MPEG2からMPEG4への変換の際にPフレームを破棄する場合とIフレームを破棄する場合がある。

## 【0072】

まず、Iフレームを破棄する場合について、図26の概念図について説明する。Pフレーム $P_{n-2}$ 及び $P_{n-1}$ 間のIフレーム $I_{n-1}$ においては動きベクトルが存在しないため、当該マクロブロックの動きベクトル $MV_n$ に捨てられたIフレームの動きベクトルを足し合わせて動きベクトルを合成することはできない。そこで、当該マクロブロックにおいては、先行例の技術で説明した方式に基づき、当該マクロブロックの動きベクトル $MV_n$ をスケーリングにより空間・時間補整を施し、次に捨てられたPフレームの前のIまたはPフレームより変換されたMPEG4のVOPを参照するように時間方向の動きベクトル $MV_n$ の長さを2倍に伸張して、動きベクトル $2 * MV_n$ を得る。これにより、当該P-VOPの動きベクトルが捨てられたフレームの直前のIまたはPフレームより変換されたVOPを参照する。このようにして、空間・時間補整によるスケーリングを行い、その後に時間方向に2倍に伸張された動きベクトルを生成し、 $8 \times 8$ の動きベクトルとして出力する。

## 【0073】

次に、Pフレームを破棄する場合について、図27、図28、図29及び図30を参照して説明する。まず、図27にPフレームを破棄する場合の処理の概念図を示す。Iフレーム $I_{n-1}$ 及びPフレーム $P_n$ 間の破棄されるPフレーム $P_{n-1}$

$MV_1$  は動きベクトル  $MV_{n-1}$  を有しているため、当該マクロブロックの動きベクトル  $MV_n$  に、破棄される P フレームの動きベクトル  $MV_{n-1}$  を足し合わせて、破棄される P フレームの前のフレームを参照する動きベクトル ( $MV_{n-1} + MV_n$ ) を生成する。

## 【0074】

図28に示すように、当該マクロブロックの動きベクトルは破棄されるフレームを参照するが、このとき参照する画素は複数のマクロブロックに重なる。

## 【0075】

図29に参照画素 ( $MB_0$ ) がどのように複数のマクロブロックに重なるかを示す。図からわかるように、参照画素 ( $MB_0$ ) は、1つ、または2つ、または4つのマクロブロック (MB) に重なる(Overlap) 場合がある。その参照画素 ( $MB_0$ ) のマクロブロック MB と重なった部分を、refMB #0、refMB #1、refMB #2、refMB #3 として示す。破棄される (Dropped) P フレームの動きベクトル及び、マクロブロックモード並びに予測モードまたはビット数、または量子化スケールは後に動きベクトルの足し合わせの合成を行うため、動きベクトル及びマクロブロック情報バッファにフレーム単位で格納する。MPEG2 からMPEG4への動きベクトル変換を行う当該マクロブロックにおいて、まず破棄された P フレーム上で当該動きベクトルが参照する座標を計算する。図29に示すように、当該動きベクトルが参照する座標によっては参照画素 ( $MB_0$ ) が破棄される参照 P フレームの1つ、または2つまたは4つのマクロブロック (MB) と重なる場合がある。複数のマクロブロックに重なる場合には予め定めたパラメータ X に基づいて、マクロブロックの符号化効率の順番を降順に並べ換える

## 【0076】

すなわち、第1の方法では、パラメータ X を「マクロブロックに重なっている画素数」とし、第2の方法では、パラメータ X を「マクロブロックの重なっている画素数／マクロブロックのビット数」とし、第3の方法では、パラメータ X を「マクロブロックの重なっている画素数／Qースケール」(Qースケールは、マクロブロック量子化スケールを意味する) とし、第4の方法では、パラメータ X

を「マクロブロックの重なっている画素数／（マクロブロックのビット数×Qースケール）とし、第5の方法では、パラメータXを（1／マクロブロックのビット数）とし、第6の方法では、パラメータXを（1／Qースケール）とし、第7の方法では、Xを1／（マクロブロックのビット数×Qースケール）とする。そして、パラメータXの最も大きいものを最大値とし、パラメータXの最も小さいものを最小値とする。図28では、第1～第4の方法におけるパラメータXを示している。図28において、MB <sub>overlapped area</sub>は、マクロブロックに重なっている画素数を示す。Coef bitsは、マクロブロックのビット数を示す。Q-scaleは、マクロブロック量子化スケールを示す。

## 【0077】

図30にPフレームを破棄する場合の動きベクトル合成アルゴリズムのフローチャートを示し、以下これについて説明する。まず、当該マクロブロックのMPEG2動きベクトルが参照する画素に重複する（重なる）、マクロブロックを構成する1つまたは2つまたは4つのマクロブロック（MB）のうち、マクロブロック（MB）モードがNot Coded（DCT係数が符号化されていない）であり、且つ、重複する（重なる）画素値が閾値T以上であるマクロブロックを検索する（ステップST-21）。閾値Tは例えば100画素とする。重複するマクロブロックのうち一つでもNot Codedのマクロブロック（MV）が含まれていれば、そのマクロブロック（MV）アドレスが最も若いNot Codedのマクロブロックの動きベクトルMV<sub>n-1</sub>を選択する（ステップST-22）。

## 【0078】

当該マクロブロックのMPEG2動きベクトルが参照する画素に重複する、マクロブロックを構成する1つまたは2つまたは4つのマクロブロックのうちNot Coded（DCT係数が符号化されていない）のマクロブロックが存在しない場合には、次に、マクロブロックモードがNo MC（動き補償なし）であり、且つ、重複する画素値が閾値T以上であるマクロブロックを検索する（ステップST-23）。閾値Tは例えば100画素とする。重複するマクロブロックのうち、一つでもNo MCのマクロブロックが含まれていれば、そのマクロブロックアドレスが最も若いNot Codedのマクロブロックの動きベクトルMV<sub>n-1</sub>を選択する（ステ

ップST-24)。

【0079】

当該マクロブロックのMPEG2動きベクトルが参照する画素に重複する(重なる)、マクロブロックを構成する1つまたは2つまたは4つのマクロブロックのうち重複するマクロブロックにNot Coded DCT係数が符号化されていない)及びNo MC(動き補償なし)が存在しない場合には、次に、上述したパラメータXが最も大きいマクロブロックより順番にマクロブロック(MB)モードがイントラマクロブロック(MB)であるかを判別する(ステップST-25)。若し、参照マクロブロックがイントラマクロブロックであれば、次にパラメータXが大きいマクロブロックがイントラマクロブロックであるかを判別し、当該マクロブロックが参照する画素に重複するマクロブロック全てがイントラマクロブロックであるかを判別する(ステップST-26)。当該マクロブロックが参照する画素に重複するマクロブロック全てがイントラマクロブロックであれば、ゼロの動きベクトル(MV)を選択する(ステップST-27)。

【0080】

ステップST-25で、イントラマクロブロックでないマクロブロックを検索した場合には、次の判別ルーチンに移る。すなわち、イントラマクロブロックでないマクロブロックにおいて、予測モードが、第2フィールドを参照するフィールド予測であるかを判別する(ステップST-28)。第2フィールドを参照するフィールド予測である場合、パラメータXが次に大きい参照マクロブロックにおいて同様の判別を行う(ステップST-29)。N(イントラマクロブロックの個数)個の参照マクロブロックが第2フィールドを参照するフィールド予測である場合には、最も大きなパラメータXをもった参照マクロブロック(MB)の動きベクトルMV<sub>n-1</sub>を選択する(ステップST-30)。Nは予め定めた数であり、1~4の値を持つ。第2フィールドを参照するフィールド予測でないマクロブロックを検索した場合には、そのマクロブロックの動きベクトルMV<sub>n-1</sub>を選択する(ステップST-31)。このようにして当該マクロブロックの動きベクトルに足し合わせる動きベクトルを選択する。

【0081】

図31に動きベクトルの第2フィールドを参照する動きベクトルの時間補整を示す。足し合わせた動きベクトルのスケーリング方法は先行例の技術で述べた動きベクトルの空間・時間補整を施すスケーリング方法と同様の方法で行う。よって、足し合わせたベクトルが参照フレームの第2フィールドを参照する場合は、動きベクトルの垂直成分に対し、フィールド補整を行い、さらに時間方向のベクトル補整として1フィールド間隔分動きベクトルを伸張する。図に示すように当該フィールドと参照する第2フィールドのフィールド間隔が $a$ であり、よって1フィールド分の時間補整を施すのに動きベクトルに $(a+1)/a$ を掛ける。このようにして $8 \times 8$ の動きベクトルを求めて出力する。

#### 【0082】

図32にIフレームを破棄する場合の動きベクトル変換装置の構成ブロック図を示す。動きベクトル変換装置は図2の動きベクトル変換装置6に示したのと同様である。まず、MPEG2の動きベクトルを動きベクトル空間・時間補整装置321に入力して、上述と同様に、動きベクトルの空間・時間補整を行い $8 \times 8$ の動きベクトルを生成する。次に $8 \times 8$ の動きベクトルを $8 \times 8$ MV再検索装置322に入力して、先行例の技術で説明したように、 $8 \times 8$ の動きベクトルの再検索処理を行う。例えば、生成された $8 \times 8$ の動きベクトルの参照先を中心に、検索サーチウインドウを整数画素で縦横2画素ずつで検索を行う。これにより $8 \times 8$ の動きベクトルの予測精度を向上させることが可能である。次に一つのマクロブロックを構成する4つの $8 \times 8$ の動きベクトルそれぞれに対して再検索された予測誤差及び、再検索処理を施した $8 \times 8$ の動きベクトルを $8 \times 8$ MV $\rightarrow$  $16 \times 16$ MV変換装置323に入力する。

#### 【0083】

$8 \times 8$ MV $\rightarrow$  $16 \times 16$ MV変換装置323では、4つの $8 \times 8$ の動きベクトルのうち、最も予測誤差が少ない動きベクトルを判別し、 $16 \times 16$ の動きベクトルに割り当てる。次に生成された $16 \times 16$ の動きベクトルを、 $16 \times 16$ MV再検索装置324に入力して、先行例の技術で述べたのと同様に再検索処理を行い、 $16 \times 16$ の動きベクトルの予測精度を向上させる。これによりMPEG4の $8 \times 8$ の動きベクトル及び、 $16 \times 16$ の動きベクトルが生成されて出力さ

れる。

【0084】

次に、図33にPフレームを破棄する場合の動きベクトル変換装置の更に他の例の構成ブロック図を示す。図33の動きベクトル変換装置は図2の動きベクトル変換装置6に示したのと同様である。まず、MPEG2の動きベクトルが入力される切換えスイッチ351は、破棄するPフレームの情報が来た場合には、MV及びMB情報バッファ352側に切換えられ、それ以外Pフレームの情報が来た場合には、参照MBアドレス計算装置354及び動きベクトル伸長空間・時間補整装置357側に切換えられる。

【0085】

MV及びMB情報バッファ352では、破棄するPフレームの動きベクトル情報及びマクロブロック予測モードをフレーム単位で格納し、場合によってはマクロブロック毎のビット量及び量子化スケールをも格納する。参照MBアドレス計算装置354においては、破棄するPフレームの次のPフレームの動きベクトルを入力とし、その当該動きベクトルの参照位置を計算する。動きベクトルの参照位置は足し合わせ動きベクトル判定装置353に入力され、参照位置を基に、参照画素に重なる破棄するPフレームのマクロブロックを算出する。足し合わせ動きベクトル判定装置353では、この情報を利用し、上述した足し合わせる動きベクトルを選択する手法に基づき処理を行い、足し合わせる動きベクトルを出力する。出力された動きベクトルは当該マクロブロックの動きベクトルと足し合わせ、合成された動きベクトルを生成する。

【0086】

次に、合成された動きベクトルは、動きベクトル空間・時間補整装置355に入力され、先行例の技術及び上段に述べた空間・時間の補整スケーリングにより変換を行って動きベクトルを生成する。動きベクトル伸長空間・時間補整装置357及び動きベクトル再検索装置358は、動きベクトルを伸張方式により求め手法であり、それぞれ動きベクトル空間・時間補整装置321及び8×8の動きベクトル再検索装置322と同様の処理を行う。これにより動きベクトルを生成する。動きベクトル再検索装置356及び動きベクトル再検索装置358より

出力された動きベクトルは、それぞれ  $8 \times 8$  の動きベクトル選択装置 359 に入力され、 $8 \times 8$  の動きベクトル選択装置 359 において 2 つの動きベクトルより予測残差の少ない動きベクトルを  $8 \times 8$  の動きベクトルとして出力する。 $8 \times 8$  MV →  $16 \times 16$  MV 動きベクトル変換装置 360 及び  $16 \times 16$  の動きベクトル再検索装置 361 の処理は、図 32 の動きベクトル変換装置における  $8 \times 8$  MV →  $16 \times 16$  MV 動きベクトル変換装置 323 及び  $16 \times 16$  の動きベクトル再検索装置 324 と同様の処理を施して、 $16 \times 16$  の動きベクトルを生成する。これにより MPEG4 の  $8 \times 8$  の動きベクトル及び  $16 \times 16$  の動きベクトルを生成して出力する。

## 【0087】

このように P フレームを破棄する場合には一つのマクロブロックを構成する 4 つのブロックそれぞれにおいて動きベクトルの伸張により求められた動きベクトル、及び動きベクトルの足し合わせにより求められた動きベクトルの 2 つが存在する。それぞれの動きベクトルに対し、次の再検索処理を施し、予測残差が少ないものを MPEG4 の  $8 \times 8$  の動きベクトルとして出力する。再検索の方法は先行例の技術で述べた方法と同様で例えばサーチウィンドウの検索範囲は縦横 2 整数画素とする。これにより、一つのマクロブロックを構成する 4 つのブロックそれぞれにおいて 2 つの動きベクトルを再検索により求めることが出来る。

## 【0088】

次に、一つのブロックにおいて、これら 2 つの動きベクトルの予測残差を比較し、予測残差の少ないものを MPEG4 の  $8 \times 8$  の動きベクトルとして出力する。このように、動きベクトルの伸張をする方式と動きベクトルの足し合わせを行う方式のうち、最も符号化効率の高い動きベクトルを選択し、 $8 \times 8$  の動きベクトルを生成することが可能である。次に、一つのマクロブロックにおいて、4 つの MPEG4  $8 \times 8$  の動きベクトル再検索より求められる予測誤差を比較し、最も予測誤差の少ない  $8 \times 8$  の動きベクトルを判定する。この最も予測誤差が少ない  $8 \times 8$  の動きベクトルを  $16 \times 16$  の動きベクトルに割り当てる。次に選択された  $16 \times 16$  の動きベクトルにおいても、上述に述べた  $8 \times 8$  の動きベクトル再検索方法と同様に再検索を行い、 $16 \times 16$  の動きベクトルの符号化効率を

最適化する。

【0089】

このように、MPEG2の画像情報圧縮情報を入力とし、MPEG4の $8 \times 8$ 及び $16 \times 16$ の動きベクトルを求める手順において、動きベクトル変換装置により、動きベクトルを伸張または足し合わせ、さらにスケーリングされた動きベクトル情報を、動きベクトルを中心とした、再検索による動きベクトル補整を行うことにより、MPEG4画像符号化装置における符号化効率の低減を最小にすることが可能である。

【0090】

次に、図34に動きベクトル変換装置の他の例の構成ブロック図を示す。図34の動きベクトル変換装置は図2の動きベクトル変換装置6に示したものと同様である。まず、MPEG2→MPEG4の $8 \times 8$ MV変換装置271では上述した動きベクトルの空間・時間補整を行い $8 \times 8$ の動きベクトルを生成する。次に、動きベクトル整数画素検索装置272において、 $8 \times 8$ の動きベクトルに先行例の技術で説明したように、 $8 \times 8$ の動きベクトルの再検索処理を行う。例えば、生成された $8 \times 8$ の動きベクトルの参照先を中心に、検索サーチウィンドウを整数画素で縦横2画素ずつで検索を行う。これにより $8 \times 8$ の動きベクトルの予測精度を向上させることが可能である。次に一つのマクロブロックを構成する4つの $8 \times 8$ の動きベクトルそれぞれに対して再検索された予測誤差及び、再検索処理を施した $8 \times 8$ の動きベクトルを、 $8 \times 8$ の動きベクトル→ $16 \times 16$ の動きベクトル変換装置273へ入力する。

【0091】

$8 \times 8$ の動きベクトル→ $16 \times 16$ の動きベクトル変換装置273では、4つの $8 \times 8$ の動きベクトルのうち、 $8 \times 8$ の動きベクトル再検索時に求められた予測残差のうち、最も予測誤差が少ない動きベクトルを判定し、 $16 \times 16$ の動きベクトルに割り当てる。次に生成された $16 \times 16$ の動きベクトルを先行例の技術で述べたのと同様に、動きベクトル整数画素検索装置274において再検索処理を行い、 $16 \times 16$ の動きベクトルの予測精度を向上させる。これによりMPEG4の $8 \times 8$ の動きベクトル及び、 $16 \times 16$ の動きベクトルを生成して出力

する。

【0092】

出力された  $8 \times 8$  の動きベクトル及び  $16 \times 16$  の動きベクトルをそれぞれ半画素検索装置 275 に入力され、半画素精度で  $8 \times 8$  及び  $16 \times 16$  の動きベクトルが出力される。

【0093】

このように、MPEG2 画像情報圧縮情報を入力とし、MPEG4 の  $8 \times 8$  及び  $16 \times 16$  の動きのベクトルを求める手順において、動きベクトル変換装置により、動きベクトルを伸張または足し合わせ、さらにスケーリングされた動きベクトル情報を、動きベクトルを中心とした、再検索による動きベクトル補整を行うことにより、MPEG4 画像符号化装置における符号化効率の低減を最小にすることが可能である。

【0094】

【発明が解決しようとする課題】

上述した図34の動きベクトル変換装置では、MPEG2 の動きベクトルを変換し、MPEG4 の動きベクトルが生成されている。しかしながら、MPEG2 のイントラフレームの動きベクトルをMPEG4 に変換する際に、動きベクトル情報を生成せずに、MPEG4 の I-VOP (MPEG4 でイントラモードで符号化されるフレームを意味する) として変換され、MPEG4 のビットストリームの I-VOP 間隔が狭く、MPEG4 の画像圧縮情報（ビットストリーム）に変換された後に符号化効率が低下するという問題点を有している。

【0095】

かかる点に鑑み、本発明は、飛び越し走査のMPEG2 の画像圧縮情報を示すビットストリームを入力とし、順次走査のMPEG4 の画像圧縮情報を示すビットストリームを出力とする画像情報変換方法における動きベクトル変換方法において、画像情報変換方法におけるMPEG4 の画像符号化の符号化効率を一層高くすることのできるものを提案しようとするものである。

【0096】

又、本発明は、飛び越し走査のMPEG2 の画像圧縮情報を示すビットストリ

ームを入力とし、順次走査のMPEG4の画像圧縮情報を示すビットストリームを出力とする画像情報変換装置における動きベクトル変換装置において、画像情報変換装置におけるMPEG4の画像符号化装置の符号化効率を一層高くするとのできるものを提案しようとするものである。

## 【0097】

## 【課題を解決するための手段】

第1の発明は、飛び越し走査のMPEG2の画像圧縮情報を示すビットストリームを入力とし、順次走査のMPEG4の画像圧縮情報を示すビットストリームを出力とする画像情報変換方法における動きベクトル変換方法において、入力となるMPEG2の画像圧縮情報を示すビットストリームにおけるMPEG2の $16 \times 16$ の動きベクトルを入力とし、MPEG4の $8 \times 8$ の動きベクトル及びMPEG4の $16 \times 16$ の動きベクトルを生成し、MPEG2のビットストリームのPフレームを一枚置きに破棄することにより、フレームレートを下げた、低ビットレートのMPEG4のビットストリームを生成する動きベクトル変換方法であって、動きベクトル補整を兼ね、入力となるMPEG2の画像圧縮情報を示すビットストリーム内のマクロブロック及びブロック毎の情報を予め格納しておき、その格納された情報を基に、破棄する直前のPフレームの動きベクトルを複製し、その複製された動きベクトルを時間方向に2倍に伸張して、MPEG2のインタラフレームより変換されたMPEG4の動きベクトルを生成するようにした動きベクトル変換方法である。

## 【0098】

第1の発明によれば、入力となるMPEG2の画像圧縮情報を示すビットストリームにおけるMPEG2の $16 \times 16$ の動きベクトルを入力とし、MPEG4の $8 \times 8$ の動きベクトル及びMPEG4の $16 \times 16$ の動きベクトルを生成し、MPEG2のビットストリームのPフレームを一枚置きに破棄することにより、フレームレートを下げた、低ビットレートのMPEG4のビットストリームを生成し、動きベクトル補整を兼ね、入力となるMPEG2の画像圧縮情報を示すビットストリーム内のマクロブロック及びブロック毎の情報を予め格納しておき、その格納された情報を基に、破棄する直前のPフレームの動きベクトルを複製し

、その複製された動きベクトルを時間方向に2倍に伸張して、MPEG2のイントラフレームより変換されたMPEG4の動きベクトルを生成する。

## 【0099】

第2の発明は、飛び越し走査のMPEG2の画像圧縮情報を示すビットストリームを入力とし、順次走査のMPEG4の画像圧縮情報を示すビットストリームを出力とする画像情報変換方法における動きベクトル変換方法において、入力となるMPEG2の画像圧縮情報を示すビットストリームにおけるMPEG2の $16 \times 16$ の動きベクトルを入力とし、MPEG4の $8 \times 8$ の動きベクトル及びMPEG4の $16 \times 16$ の動きベクトルを生成し、MPEG2のビットストリームのIフレーム又はPフレームを一枚置きに破棄することにより、フレームレートを下げた、低ビットレートのMPEG4のビットストリームを生成する動きベクトル変換方法であって、MPEG2のイントラフレームより変換されたMPEG4の動きベクトルの符号化効率を判定するパラメータとして、入力となる直前に変換されたMPEG2の画像圧縮情報を示すビットストリームにおけるマクロブロック及びブロック毎の動きベクトル情報をを利用して、動きベクトル0の予測との比較を行い、予測残差の少ない動きベクトルを符号化効率の高いものとして選択するようにした動きベクトル変換方法である。

## 【0100】

第3の発明は、飛び越し走査のMPEG2の画像圧縮情報を示すビットストリームを入力とし、順次走査のMPEG4の画像圧縮情報を示すビットストリームを出力とする画像情報変換方法における動きベクトル変換方法において、入力となるMPEG2の画像圧縮情報を示すビットストリームにおけるMPEG2の $16 \times 16$ の動きベクトルを入力とし、MPEG4の $8 \times 8$ の動きベクトル及びMPEG4の $16 \times 16$ の動きベクトルを生成し、MPEG2のビットストリームのIフレーム又はPフレームを一枚置きに破棄することにより、フレームレートを下げた、低ビットレートのMPEG4のビットストリームを生成する動きベクトル変換方法であって、動きベクトル補整を兼ね、入力となるMPEG2の画像圧縮情報を示すビットストリーム内の動きベクトル情報を基に、MPEG2の動きベクトルより動きベクトル変換方法により変換されたMPEG4の $8 \times 8$ の動

きベクトルを入力とし、その入力動きベクトル値を中心とした動きベクトルの再検索を行って、動きベクトルを補整することにより、一つのマクロブロックを構成する4つの $8 \times 8$ の動きベクトルのうち、最も予測残差の少ないMPEG4の $8 \times 8$ の動きベクトルを $16 \times 16$ の動きベクトルに割り当てるにより、 $16 \times 16$ の動きベクトルを生成するようにした動きベクトル変換方法である。

#### 【0101】

第4の発明は、飛び越し走査のMPEG2の画像圧縮情報を示すビットストリームを入力とし、順次走査のMPEG4の画像圧縮情報を示すビットストリームを出力とする画像情報変換装置における動きベクトル変換装置において、入力となるMPEG2の画像圧縮情報を示すビットストリームにおけるMPEG2の $16 \times 16$ の動きベクトルを入力とし、MPEG4の $8 \times 8$ の動きベクトル及びMPEG4の $16 \times 16$ の動きベクトルを生成する動きベクトル変換装置であって、MPEG2のビットストリームのPフレームを一枚置きに破棄する破棄手段を設け、その破棄手段によって、フレームレートを下げた、低ビットレートのMPEG4のビットストリームを生成すると共に、動きベクトル補整装置を兼ね備え、入力となるMPEG2の画像圧縮情報を示すビットストリーム内のマクロブロック及びブロック毎の情報を予め格納しておく格納手段と、その格納手段に格納された情報を基に、破棄する直前のPフレームの動きベクトルを複製し、その複製された動きベクトルを時間方向に2倍に伸張して、MPEG2のイントラフレームより変換されたMPEG4の動きベクトルを生成する動きベクトル生成手段とを有する動きベクトル変換装置である。

#### 【0102】

第5の発明は、飛び越し走査のMPEG2の画像圧縮情報を示すビットストリームを入力とし、順次走査のMPEG4の画像圧縮情報を示すビットストリームを出力とする画像情報変換装置における動きベクトル変換装置において、入力となるMPEG2の画像圧縮情報を示すビットストリームにおけるMPEG2の $16 \times 16$ の動きベクトルを入力とし、MPEG4の $8 \times 8$ の動きベクトル及びMPEG4の $16 \times 16$ の動きベクトルを生成する動きベクトル変換装置であって、MPEG2のビットストリームのIフレーム又はPフレームを一枚置きに破棄

する破棄手段を設け、その破棄手段によって、フレームレートを下げた、低ビットレートのMPEG4のビットストリームを生成すると共に、MPEG2のイントラフレームより変換されたMPEG4の動きベクトルの符号化効率を判定するパラメータとして、入力となる直前に変換されたMPEG2の画像圧縮情報を示すビットストリームにおけるマクロブロック及びブロック毎の動きベクトル情報をを利用して、動きベクトル0の予測との比較を行い、予測残差の少ない動きベクトルを符号化効率の高いものとして選択する選択手段を有する動きベクトル変換装置である。

## 【0103】

第6の発明は、飛び越し走査のMPEG2の画像圧縮情報を示すビットストリームを入力とし、順次走査のMPEG4の画像圧縮情報を示すビットストリームを出力とする画像情報変換装置における動きベクトル変換装置において、入力となるMPEG2の画像圧縮情報を示すビットストリームにおけるMPEG2の $16 \times 16$ の動きベクトルを入力とし、MPEG4の $8 \times 8$ の動きベクトル及びMPEG4の $16 \times 16$ の動きベクトルを生成する動きベクトル変換装置であって、MPEG2のビットストリームのIフレーム又はPフレームを一枚置きに破棄する破棄手段を設け、その破棄手段によって、フレームレートを下げた、低ビットレートのMPEG4のビットストリームを生成すると共に、動きベクトル補整装置を兼ね備え、入力となるMPEG2の画像圧縮情報を示すビットストリーム内の動きベクトル情報を基に、MPEG2の動きベクトルより動きベクトル変換装置により変換されたMPEG4の $8 \times 8$ の動きベクトルを入力とし、その入力動きベクトル値を中心とした動きベクトルの再検索を行って、動きベクトルを補整することにより、一つのマクロブロックを構成する4つの $8 \times 8$ の動きベクトルのうち、最も予測残差の少ないMPEG4の $8 \times 8$ の動きベクトルを $16 \times 16$ の動きベクトルに割り当てるにより、 $16 \times 16$ の動きベクトルを生成する動きベクトル生成手段を有する動きベクトル変換装置である。

## 【0104】

## 【発明の実施の形態】

以下、図面を参照して、本発明の実施の形態の動きベクトル変換方法及び変換

装置の例を説明する。図35を参照して、イントラフレームから変換されるVOPをP-VOPに変換する方法を説明する。当該フレームがイントラフレームである場合、動きベクトルを所有しないため、画像符号化に動き補償予測を利用できず、符号化効率が下がる。そこで、動き補償予測用に動きベクトルを生成するために、破棄する直前のPフレームの動きベクトルをバッファに格納し、上述した空間・時間補整用のスケーリングを行い、当該フレームの同位置のマクロブロックにそれぞれ複製する。次に、直前のPフレームは破棄されるため、参照フレームとの時間間隔が2倍になる。よって、複製した動きベクトルを時間方向に2倍に伸張し、破棄するPフレームの直前のPフレームを参照するように補整を行う。

#### 【0105】

図36に動きベクトル変換装置のブロック構成図を示す。図2の動きベクトル変換装置6に相当する動きベクトル変換装置部を示したものが図36のブロック図である。スイッチ361はI→P-VOP動きベクトル(MV)変換フラグにより動作し、MPEG4に変換するMPEG2イントラフレーム及びその直前のMPEG2がPフレームの時には、I→P-VOP動きベクトル変換装置362側に切り換えられ、それ以外のフレームでは、P→P-VOP動きベクトル変換装置363側に切り換えられる。

#### 【0106】

I→P-VOP変換装置362については、後に説明する。P→P-VOP変換装置363は、図1の従来の技術で説明した動きベクトル変換装置と同様の処理を行う。

#### 【0107】

図37にI→P-VOP動きベクトル変換装置の構成ブロック図を示す。まず、MPEG2の動きベクトルバッファ371にIフレームの破棄する直前のPフレームの動きベクトルを格納する。Iフレームのマクロブロックにそれぞれ破棄するPフレーム内の同位置のマクロブロックの動きベクトルを複製する。MPEG2→MPEG4の8×8のMV変換装置372においては、上述の技術で述べた動きベクトルの空間・時間補整を行った後に、Pフレームの破棄する直前のP

フレームを参照するように時間方向に動きベクトルを2倍に伸張する。変換されたMPEG4の $8 \times 8$ 動きベクトルは、0MVと予測誤差の比較装置373に入力され、動きベクトルが0の予測と、予測残差とを比較し、0動きベクトルの予測残差が生成した $8 \times 8$ の動きベクトルの予測残差より小さい場合には、 $8 \times 8$ の動きベクトルを0に置き換える。これにより出力された $8 \times 8$ の動きベクトルは、動きベクトル整数画素検索装置374において、上述の技術で説明した方法と同様に、整数画素値で指定された任意のサーチウィンドウ内で再検索を行い、動きベクトルの予測精度を向上させる。

## 【0108】

そして、 $8 \times 8$ MV $\rightarrow$  $16 \times 16$ MV変換装置375において、生成された $8 \times 8$ 動きベクトル及びそれぞれの予測残差を入力とし、一つのマクロブロックを構成する4つのブロックのうち、最も予測残差が少なかった $8 \times 8$ の動きベクトルを $16 \times 16$ の動きベクトルに割り当てて出力する。0MVと予測誤差の比較装置376は、生成された $16 \times 16$ の動きベクトルを入力とし、0動きベクトルの予測残差と比較し、0動きベクトルの予測残差が生成された $16 \times 16$ の動きベクトルより予測残差が少なければ、 $16 \times 16$ の動きベクトルを0に置き換える。これにより生成された $16 \times 16$ の動きベクトルを動きベクトル整数画素検索装置377の入力とし、動きベクトル整数画素検索装置377では $16 \times 1$ の動きベクトルを上述の技術で説明した方法に基づいて、整数画素で再検索を行って出力する。整数画素値で求められた $8 \times 8$ 及び $16 \times 16$ の動きベクトルは、それぞれ動きベクトル半画素検索装置378に入力され、半画素精度でMPEG4の $8 \times 8$ 及び $16 \times 16$ の動きベクトルを出力する。

## 【0109】

以上、入力としてMPEG2の画像圧縮情報を示すビットストリームを、出力としてMPEG4の画像圧縮情報を示すビットストリームを対象としてきたが、入力、出力ともこれに限らず、例えばMPEG-1やH.263などの画像圧縮情報（ビットストリーム）でも良い。

## 【0110】

## 【発明の効果】

第1の発明は、飛び越し走査のMPEG2の画像圧縮情報を示すビットストリームを入力とし、順次走査のMPEG4の画像圧縮情報を示すビットストリームを出力とする画像情報変換方法における動きベクトル変換方法において、入力となるMPEG2の画像圧縮情報を示すビットストリームにおけるMPEG2の $16 \times 16$ の動きベクトルを入力とし、MPEG4の $8 \times 8$ の動きベクトル及びMPEG4の $16 \times 16$ の動きベクトルを生成し、MPEG2のビットストリームのPフレームを一枚置きに破棄することにより、フレームレートを下げた、低ビットレートのMPEG4のビットストリームを生成する動きベクトル変換方法であって、動きベクトル補整を兼ね、入力となるMPEG2の画像圧縮情報を示すビットストリーム内のマクロブロック及びブロック毎の情報を予め格納しておき、その格納された情報を基に、破棄する直前のPフレームの動きベクトルを複製し、その複製された動きベクトルを時間方向に2倍に伸張して、MPEG2のインタラフレームより変換されたMPEG4の動きベクトルを生成するようにした動きベクトル変換方法である。

#### 【0111】

第2の発明は、飛び越し走査のMPEG2の画像圧縮情報を示すビットストリームを入力とし、順次走査のMPEG4の画像圧縮情報を示すビットストリームを出力とする画像情報変換方法における動きベクトル変換方法において、入力となるMPEG2の画像圧縮情報を示すビットストリームにおけるMPEG2の $16 \times 16$ の動きベクトルを入力とし、MPEG4の $8 \times 8$ の動きベクトル及びMPEG4の $16 \times 16$ の動きベクトルを生成し、MPEG2のビットストリームのIフレーム又はPフレームを一枚置きに破棄することにより、フレームレートを下げた、低ビットレートのMPEG4のビットストリームを生成する動きベクトル変換方法であって、MPEG2のインタラフレームより変換されたMPEG4の動きベクトルの符号化効率を判定するパラメータとして、入力となる直前に変換されたMPEG2の画像圧縮情報を示すビットストリームにおけるマクロブロック及びブロック毎の動きベクトル情報をを利用して、動きベクトル0の予測との比較を行い、予測残差の少ない動きベクトルを符号化効率の高いものとして選択するようにした動きベクトル変換方法である。

## 【0112】

第3の発明は、飛び越し走査のMPEG2の画像圧縮情報を示すビットストリームを入力とし、順次走査のMPEG4の画像圧縮情報を示すビットストリームを出力とする画像情報変換方法における動きベクトル変換方法において、入力となるMPEG2の画像圧縮情報を示すビットストリームにおけるMPEG2の $16 \times 16$ の動きベクトルを入力とし、MPEG4の $8 \times 8$ の動きベクトル及びMPEG4の $16 \times 16$ の動きベクトルを生成し、MPEG2のビットストリームのIフレーム又はPフレームを一枚置きに破棄することにより、フレームレートを下げた、低ビットレートのMPEG4のビットストリームを生成する動きベクトル変換方法であって、動きベクトル補整を兼ね、入力となるMPEG2の画像圧縮情報を示すビットストリーム内の動きベクトル情報を基に、MPEG2の動きベクトルより動きベクトル変換方法により変換されたMPEG4の $8 \times 8$ の動きベクトルを入力とし、その入力動きベクトル値を中心とした動きベクトルの再検索を行って、動きベクトルを補整することにより、一つのマクロブロックを構成する4つの $8 \times 8$ の動きベクトルのうち、最も予測残差の少ないMPEG4の $8 \times 8$ の動きベクトルを $16 \times 16$ の動きベクトルに割り当てるこにより、 $16 \times 16$ の動きベクトルを生成するようにした動きベクトル変換方法である。

## 【0113】

第1、第2及び第3の発明によれば、画像情報変換方法におけるMPEG4の画像符号化の符号化効率を一層高くすることのできる動きベクトル変換方法を得ることができる。

## 【0114】

第4の発明は、飛び越し走査のMPEG2の画像圧縮情報を示すビットストリームを入力とし、順次走査のMPEG4の画像圧縮情報を示すビットストリームを出力とする画像情報変換装置における動きベクトル変換装置において、入力となるMPEG2の画像圧縮情報を示すビットストリームにおけるMPEG2の $16 \times 16$ の動きベクトルを入力とし、MPEG4の $8 \times 8$ の動きベクトル及びMPEG4の $16 \times 16$ の動きベクトルを生成する動きベクトル変換装置であって、MPEG2のビットストリームのPフレームを一枚置きに破棄する破棄手段を

設け、その破棄手段によって、フレームレートを下げた、低ビットレートのMPEG4のビットストリームを生成すると共に、動きベクトル補整装置を兼ね備え、入力となるMPEG2の画像圧縮情報を示すビットストリーム内のマクロブロック及びブロック毎の情報を予め格納しておく格納手段と、その格納手段に格納された情報を基に、破棄する直前のPフレームの動きベクトルを複製し、その複製された動きベクトルを時間方向に2倍に伸張して、MPEG2のイントラフレームより変換されたMPEG4の動きベクトルを生成する動きベクトル生成手段とを有する動きベクトル変換装置である。

## 【0115】

第5の発明は、飛び越し走査のMPEG2の画像圧縮情報を示すビットストリームを入力とし、順次走査のMPEG4の画像圧縮情報を示すビットストリームを出力とする画像情報変換装置における動きベクトル変換装置において、入力となるMPEG2の画像圧縮情報を示すビットストリームにおけるMPEG2の $16 \times 16$ の動きベクトルを入力とし、MPEG4の $8 \times 8$ の動きベクトル及びMPEG4の $16 \times 16$ の動きベクトルを生成する動きベクトル変換装置であって、MPEG2のビットストリームのIフレーム又はPフレームを一枚置きに破棄する破棄手段を設け、その破棄手段によって、フレームレートを下げた、低ビットレートのMPEG4のビットストリームを生成すると共に、MPEG2のイントラフレームより変換されたMPEG4の動きベクトルの符号化効率を判定するパラメータとして、入力となる直前に変換されたMPEG2の画像圧縮情報を示すビットストリームにおけるマクロブロック及びブロック毎の動きベクトル情報をを利用して、動きベクトル0の予測との比較を行い、予測残差の少ない動きベクトルを符号化効率の高いものとして選択する選択手段を有する動きベクトル変換装置である。

## 【0116】

第6の発明は、飛び越し走査のMPEG2の画像圧縮情報を示すビットストリームを入力とし、順次走査のMPEG4の画像圧縮情報を示すビットストリームを出力とする画像情報変換装置における動きベクトル変換装置において、入力となるMPEG2の画像圧縮情報を示すビットストリームにおけるMPEG2の1

6×16の動きベクトルを入力とし、MPEG4の8×8の動きベクトル及びMPEG4の16×16の動きベクトルを生成する動きベクトル変換装置であって、MPEG2のビットストリームのIフレーム又はPフレームを一枚置きに破棄する破棄手段を設け、その破棄手段によって、フレームレートを下げた、低ビットレートのMPEG4のビットストリームを生成すると共に、動きベクトル補整装置を兼ね備え、入力となるMPEG2の画像圧縮情報を示すビットストリーム内の動きベクトル情報を基に、MPEG2の動きベクトルより動きベクトル変換装置により変換されたMPEG4の8×8の動きベクトルを入力とし、その入力動きベクトル値を中心とした動きベクトルの再検索を行って、動きベクトルを補整することにより、一つのマクロブロックを構成する4つの8×8の動きベクトルのうち、最も予測残差の少ないMPEG4の8×8の動きベクトルを16×16の動きベクトルに割り当てるにより、16×16の動きベクトルを生成する動きベクトル生成手段を有する動きベクトル変換装置である。

## 【0117】

第4、第5及び第6の発明によれば、画像情報変換方法におけるMPEG4の画像符号化の符号化効率を一層高くすることのできる動きベクトル変換装置を得ることができる。

## 【図面の簡単な説明】

## 【図1】

MPEG2の画像圧縮情報を示すビットストリームからMPEG4の画像情報圧縮装置を示すビットストリームへの変換を実現する画像情報変換装置の従来例を示すブロック線図である。

## 【図2】

筆者らの発明によるMPEG2の圧縮画像情報を示すビットストリームからMPEG4の圧縮画像情報を示すビットストリームへの変換を実現する画像情報変換装置（先行例）の構成を示すブロック線図である。

## 【図3】

MPEG2の画像圧縮情報を示すビットストリーム中の動きベクトルと、MPEG4の画像圧縮情報を示すビットストリーム中の動きベクトルの相関を示す説

明図である。

【図4】

図2の画像情報変換装置における解像度フレームレート変換装置5における動作原理を示す説明図である。

【図5】

図2の画像情報変換装置における画枠サイズ調整フラグに応じた解像度フレームレート変換装置5における画素の補填或いは除去の動作原理を示す説明図である。

【図6】

図2の画像情報変換装置における動きベクトル変換装置6における動きベクトル変換方法を示す説明図である。

【図7】

図2の画像情報変換装置における動きベクトル変換装置6の具体構成を示すブロック線図である。

【図8】

図2の画像情報変換装置におけるMPEG2の $16 \times 16$ の動きベクトル→MPEG4の $8 \times 8$ の動きベクトル変換装置8における動作原理を示すフローチャートである。

【図9】

図7の動きベクトル変換装置におけるMPEG2の $16 \times 16$ の動きベクトル→MPEG4の $8 \times 8$ の動きベクトル変換装置8でのフレーム構造、フレーム予測の場合の動きベクトル変換の概念を示す説明図である。

【図10】

図9におけるMPEG2の画像圧縮情報を示すビットストリーム中の半画素精度の動きベクトルの、MPEG4の $8 \times 8$ の動きベクトルへの変換後の取り扱いを示す表図である。

【図11】

図9における画像がフレーム構造で、第一フィールド予測であるときの動きベクトル変換の概念を示す説明図である。

【図12】

図9における画像がフレーム構造で、第二フィールド予測であるときについての動きベクトル変換の概念を示す説明図である。

【図13】

図9における画像がフレーム構造で、第二フィールド予測であるときについての動きベクトル変換の概念を示す説明図である。

【図14】

図7の動きベクトル変換装置における画像サイズ調整フラグによる動きベクトル調整器9の動作を示すフローチャートである。

【図15】

図7の動きベクトルにおけるMPEG2のイントラマクロブロックに対する動きベクトル補整器11の動作原理を示すブロック線図である。

【図16】

図2の画像情報変換装置における動きベクトル変換装置6の具体構成を示すブロック線図である。

【図17】

図7の動きベクトル変換装置における動きベクトル再検索を行うMPEG2の $16 \times 16$ の動きベクトル→MPEG4の $8 \times 8$ の動きベクトル変換装置8でのフレーム構造、フレーム予測の場合の動きベクトル変換の概念を示す説明図である。

【図18】

図7の動きベクトル変換装置における動きベクトル再検索を行う場合のMPEG2の画像圧縮情報を示すビットストリーム中の半画素精度の動きベクトルの、MPEG4 $8 \times 8$ の動きベクトルへの変換後の取り扱いを示した表図である。

【図19】

図7の動きベクトル変換装置における動きベクトル再検索を行う場合の画像がフレーム構造で、第一フィールド予測である時について動きベクトル変換の概念を示す説明図である。

【図20】

図7の動きベクトル変換装置における動きベクトル再検索を行う場合の画像がフレーム構造で、第二フィールド予測であるときの動きベクトル変換の概念を示す説明図である。

【図21】

動きベクトル補整装置を示すブロック線図である。

【図22】

$8 \times 8$  動きベクトルに施し、予測残差に基づき  $16 \times 16$  の動きベクトルを生成する方式の動きベクトル再検索装置を示すブロック線図である。

【図23】

空間補整による動きベクトルの歪を示す説明図である。

【図24】

空間補整による動きベクトルの歪を示す説明図である。

【図25】

フレーム破棄の説明図である。

【図26】

I フレーム破棄の説明図である。

【図27】

P フレーム破棄の説明図である。

【図28】

P フレーム破棄の説明図である。

【図29】

参照画素がどのように複数のマクロブロックに重なるかを示す説明図である。

【図30】

P フレームを破棄する場合の動きベクトル合成アルゴリズムを示すフローチャートである。

【図31】

動きベクトルの第2フィールドを参照する動きベクトルの時間補整の説明図である。

【図32】

動きベクトル変換装置を示すブロック線図である。

【図33】

動きベクトル変換装置を示すブロック線図である。

【図34】

動きベクトル変換装置を示すブロック線図である。

【図35】

本発明の実施の形態の動きベクトル変換方法の例のイントラフレームから変換されるVOPをP-VOPに変換する方法を示す説明図である。

【図36】

本発明の実施の形態の動きベクトル変換装置の例を示すブロック線図である。

【図37】

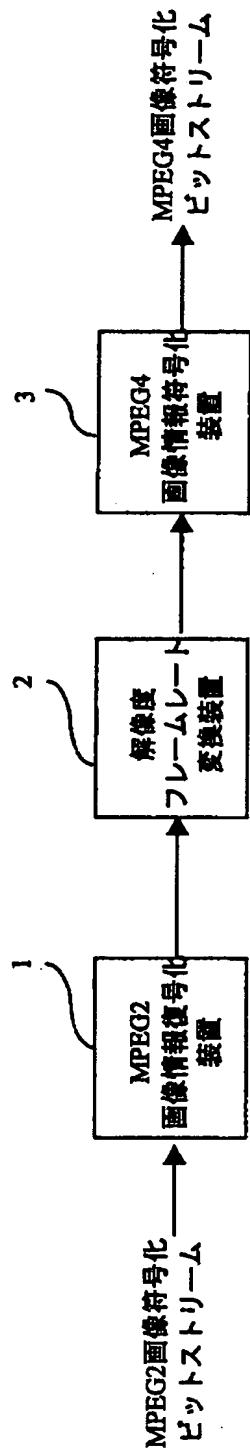
本発明の実施の形態のI→P動きベクトル変換装置の例を示すブロック線図である。

【符号の説明】

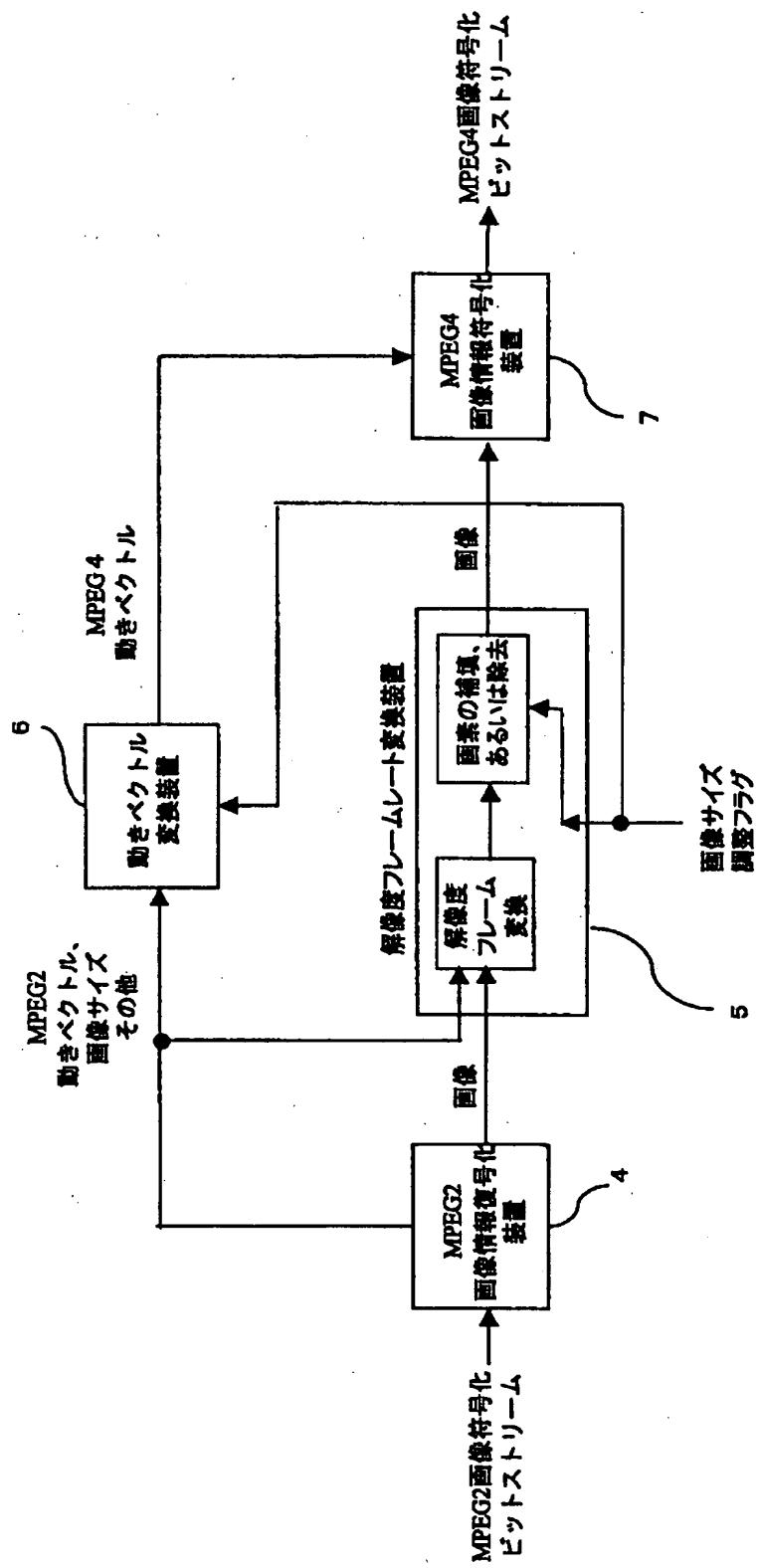
361 切り替えスイッチ、362 I→P-VOP動きベクトル変換装置、  
363 P→P-VOP動きベクトル変換装置、371 MPEG2動きベクトルバッファ、  
372 MPEG2→MPEG4 8×8MV変換装置、373  
0MVと予測誤差の比較装置、374 動きベクトル整数画素検索装置、375  
8×8MV→16×16MV変換装置、376 0MVと予測誤差の比較装置  
、377 動きベクトル整数画素検索装置、378 動きベクトル半画素検索裝置。

【書類名】 図面

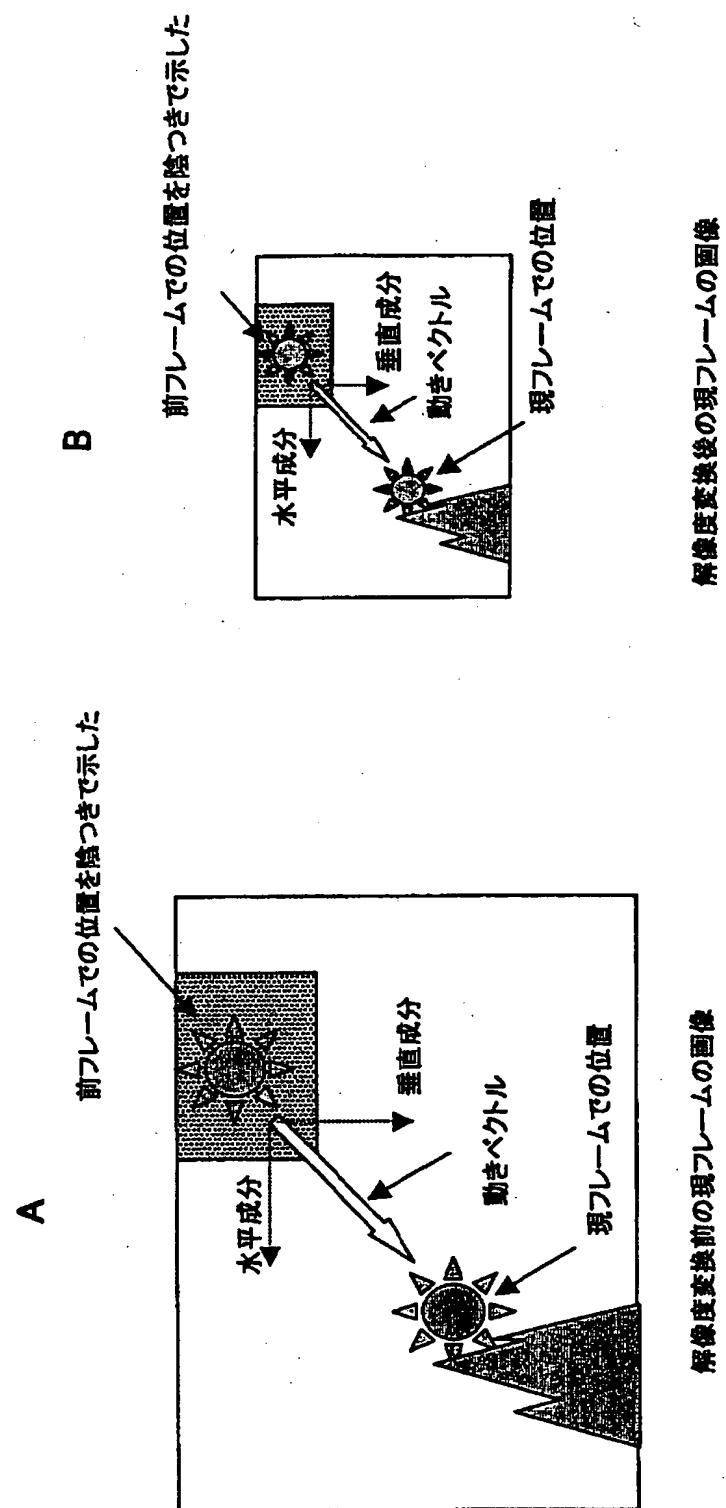
【図1】



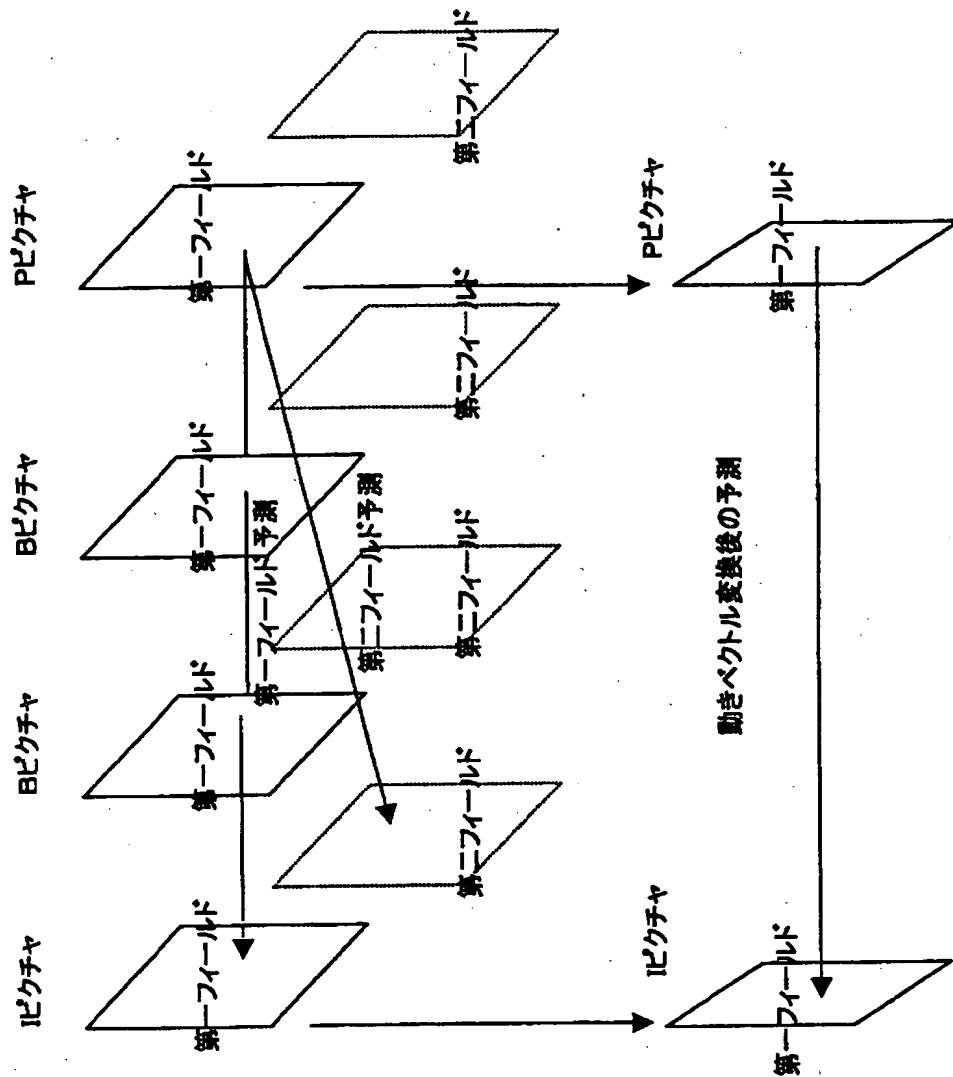
【図2】



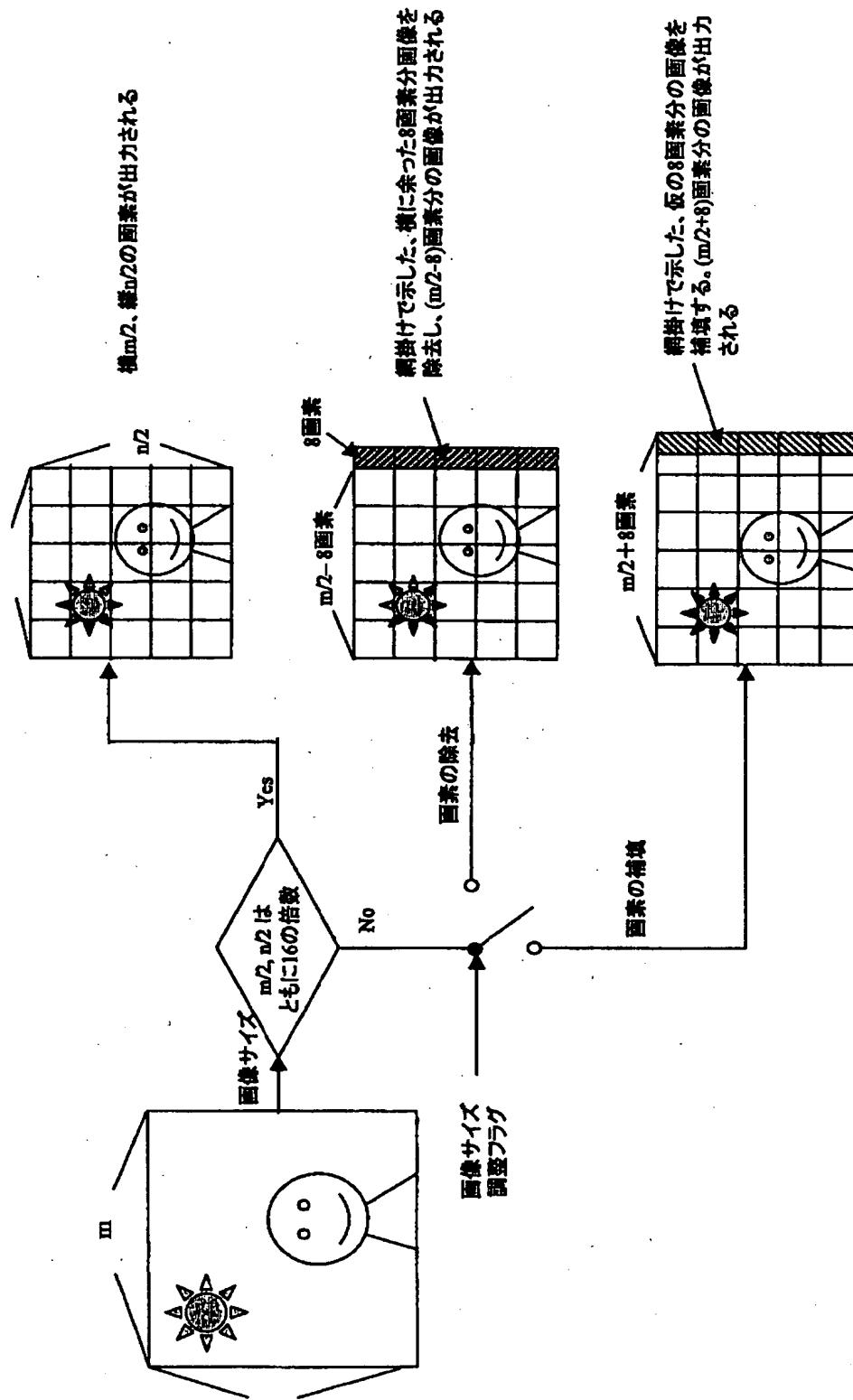
【図3】



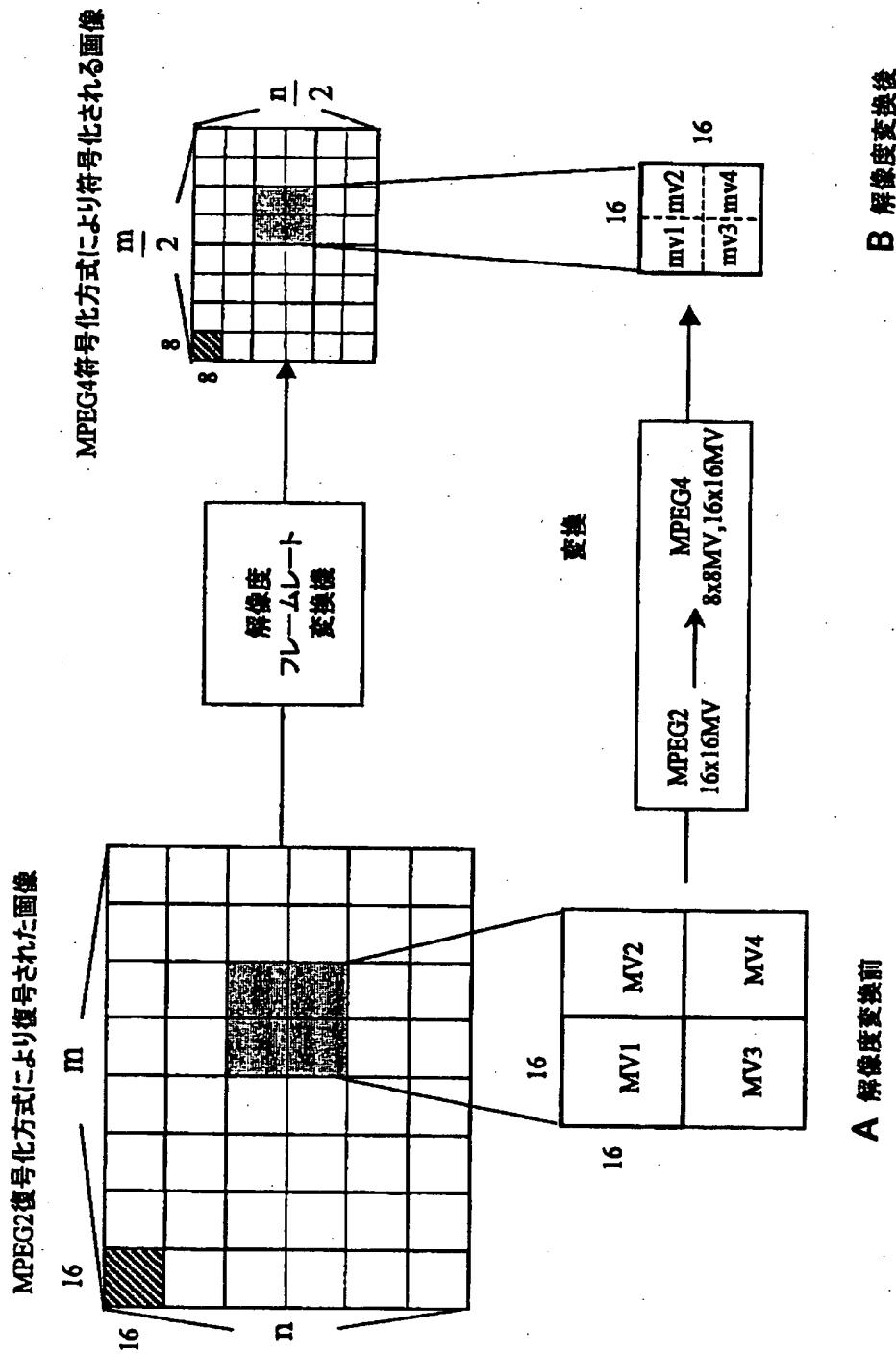
【図4】



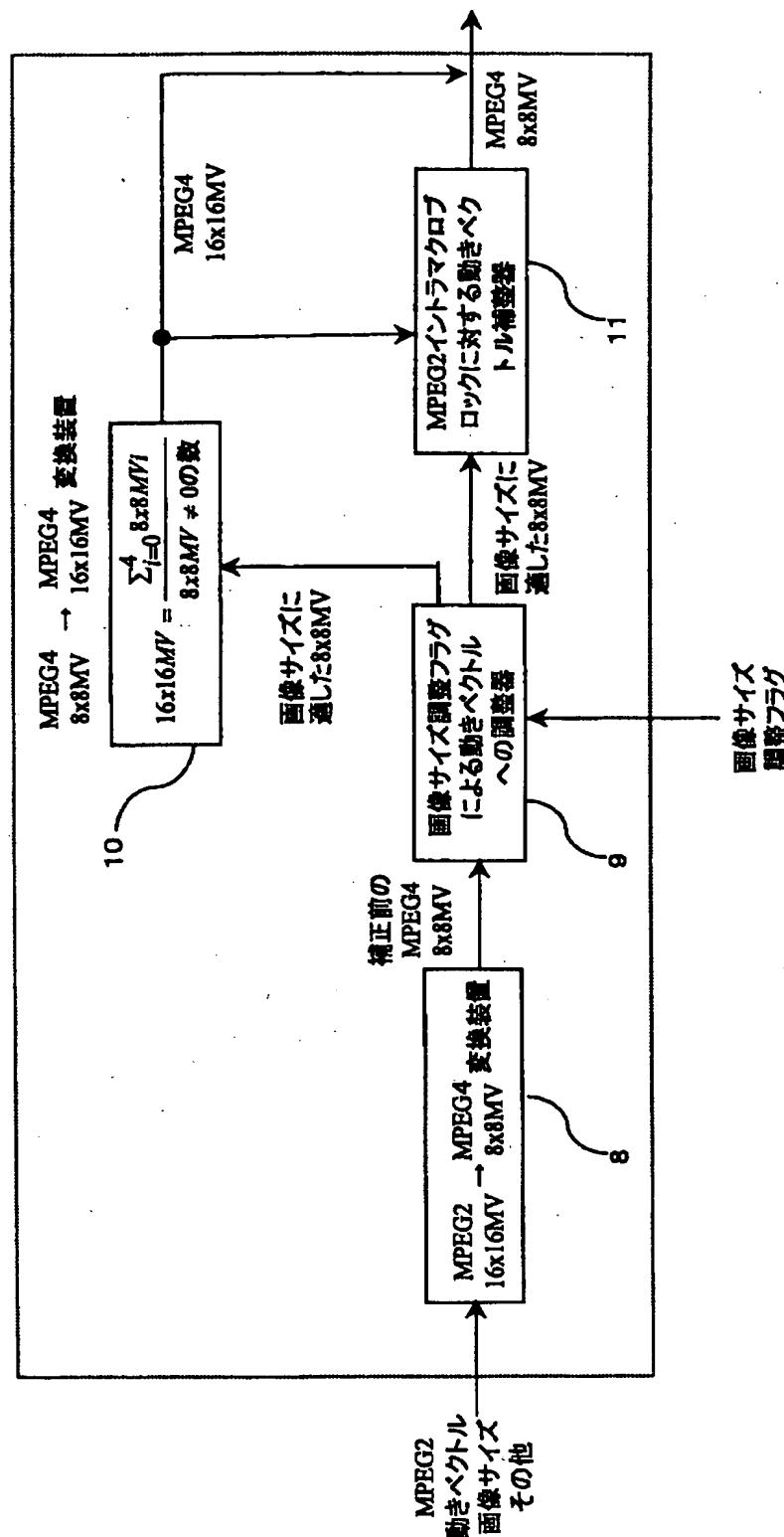
【図5】



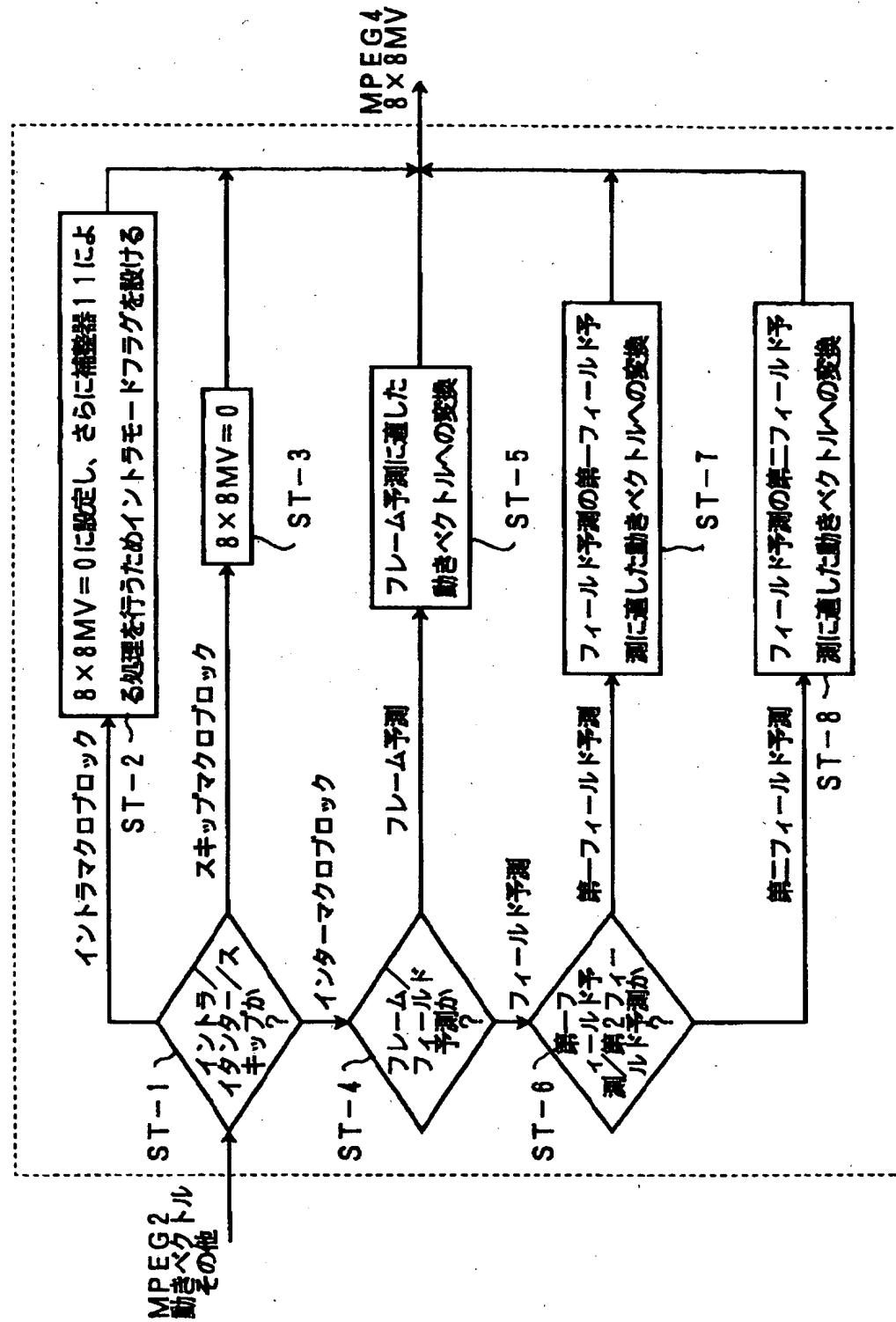
【図6】



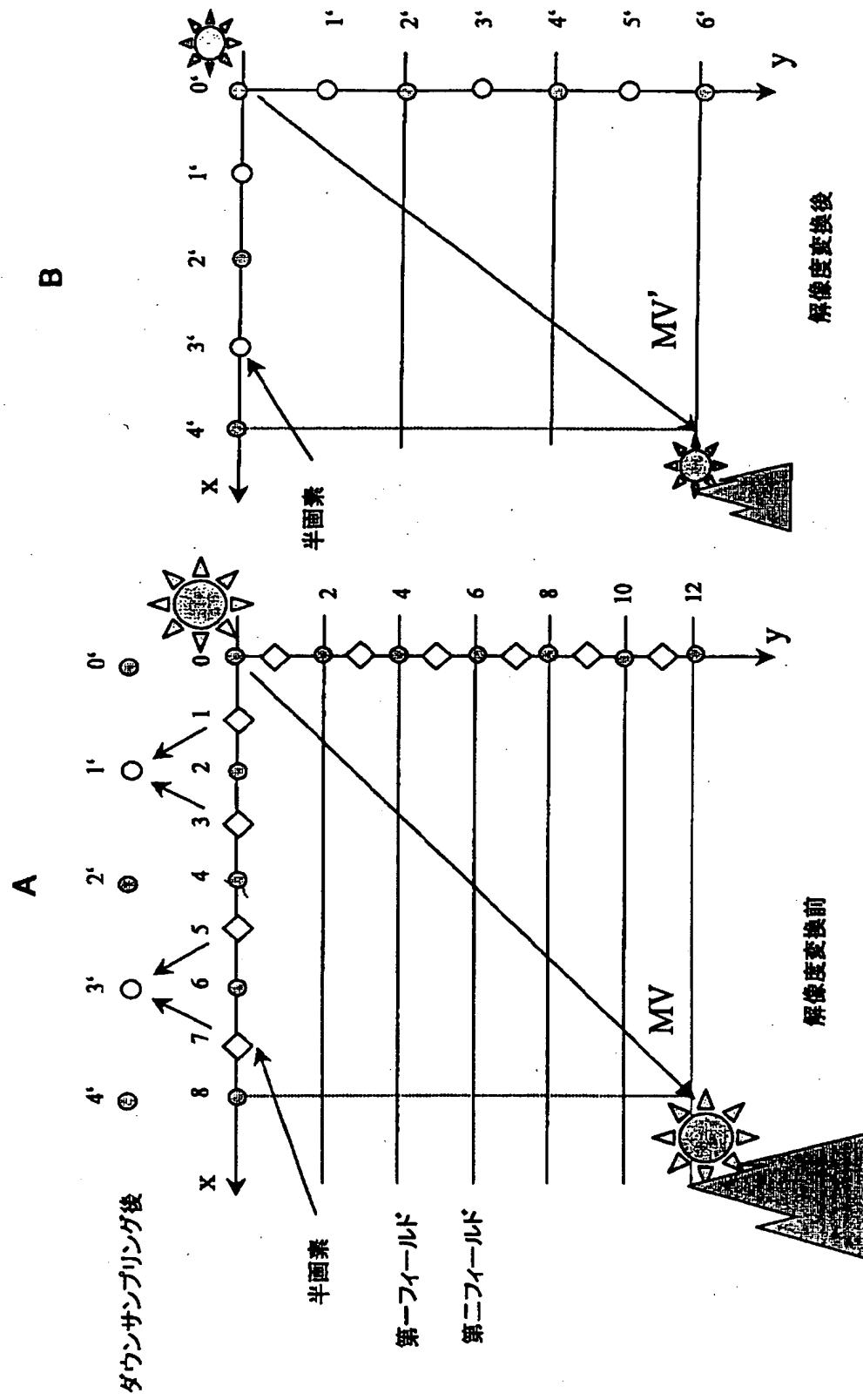
【図7】



【図8】



【図9】

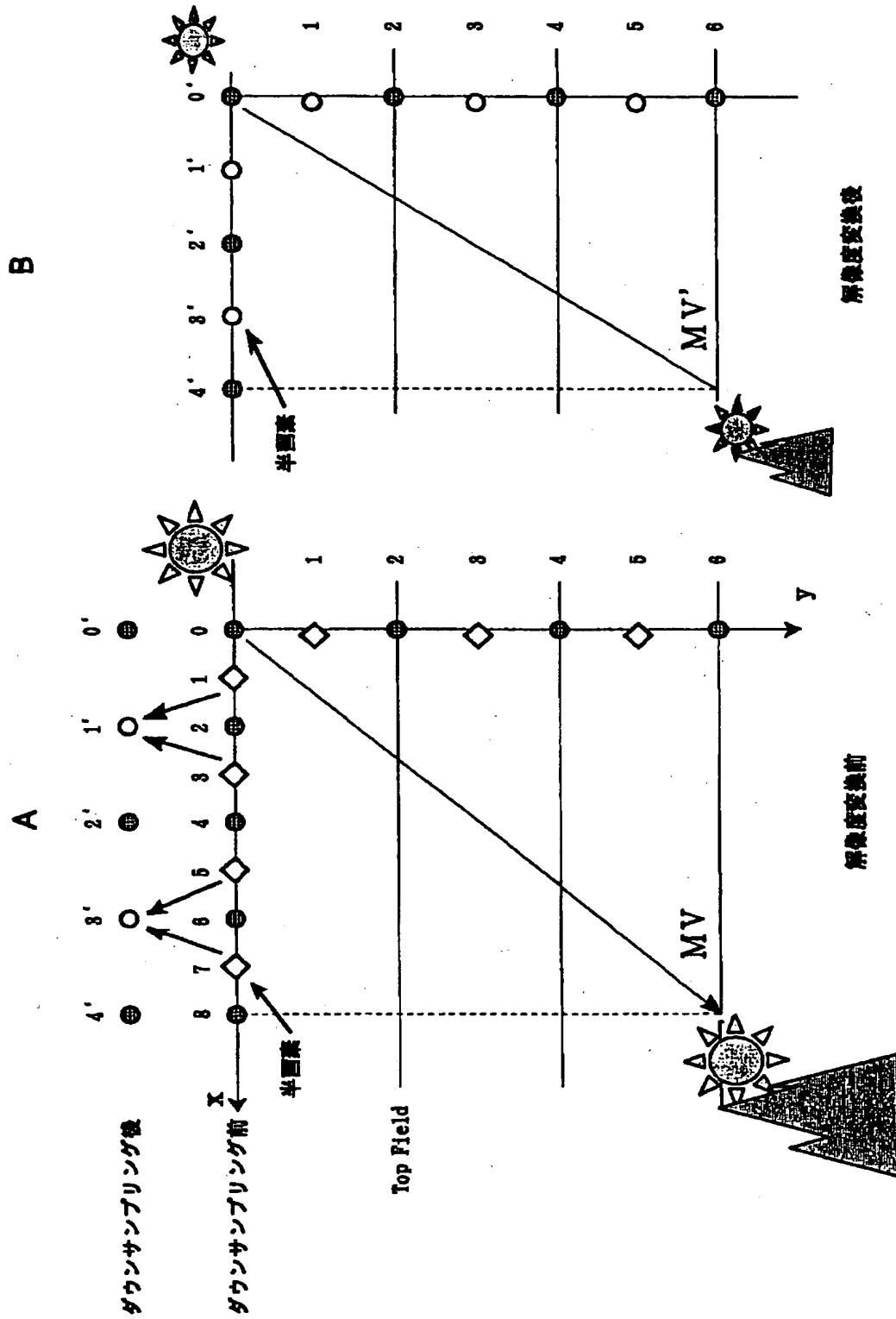


【図10】

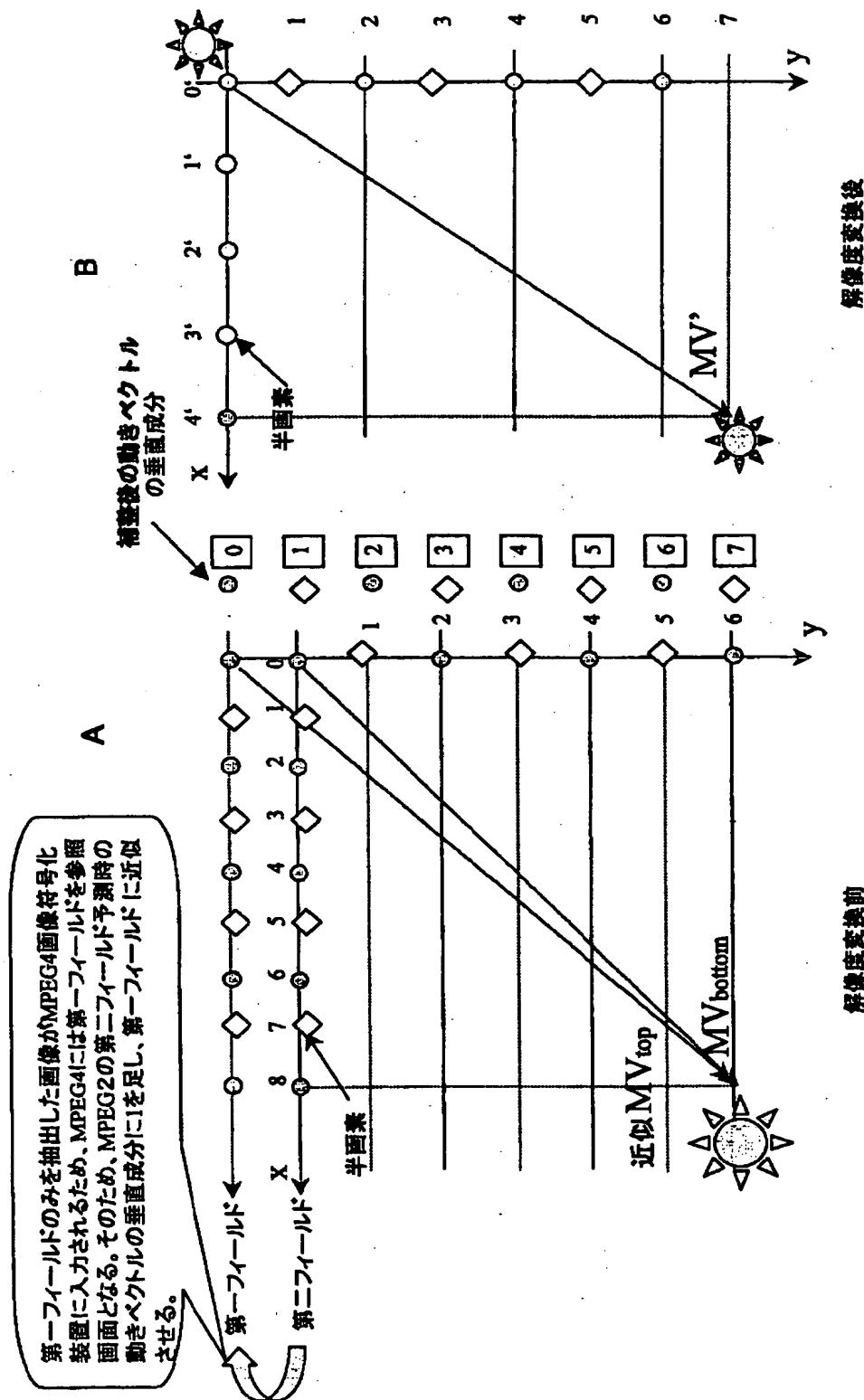
変換前の動きベクトルMV を4で割った余り	0	1	2	3
変換後の動きベクトル	[MV/2]	[MV/2]+1	[MV/2]	[MV/2]

[MV/2]はMVを2で割った整数部を表す

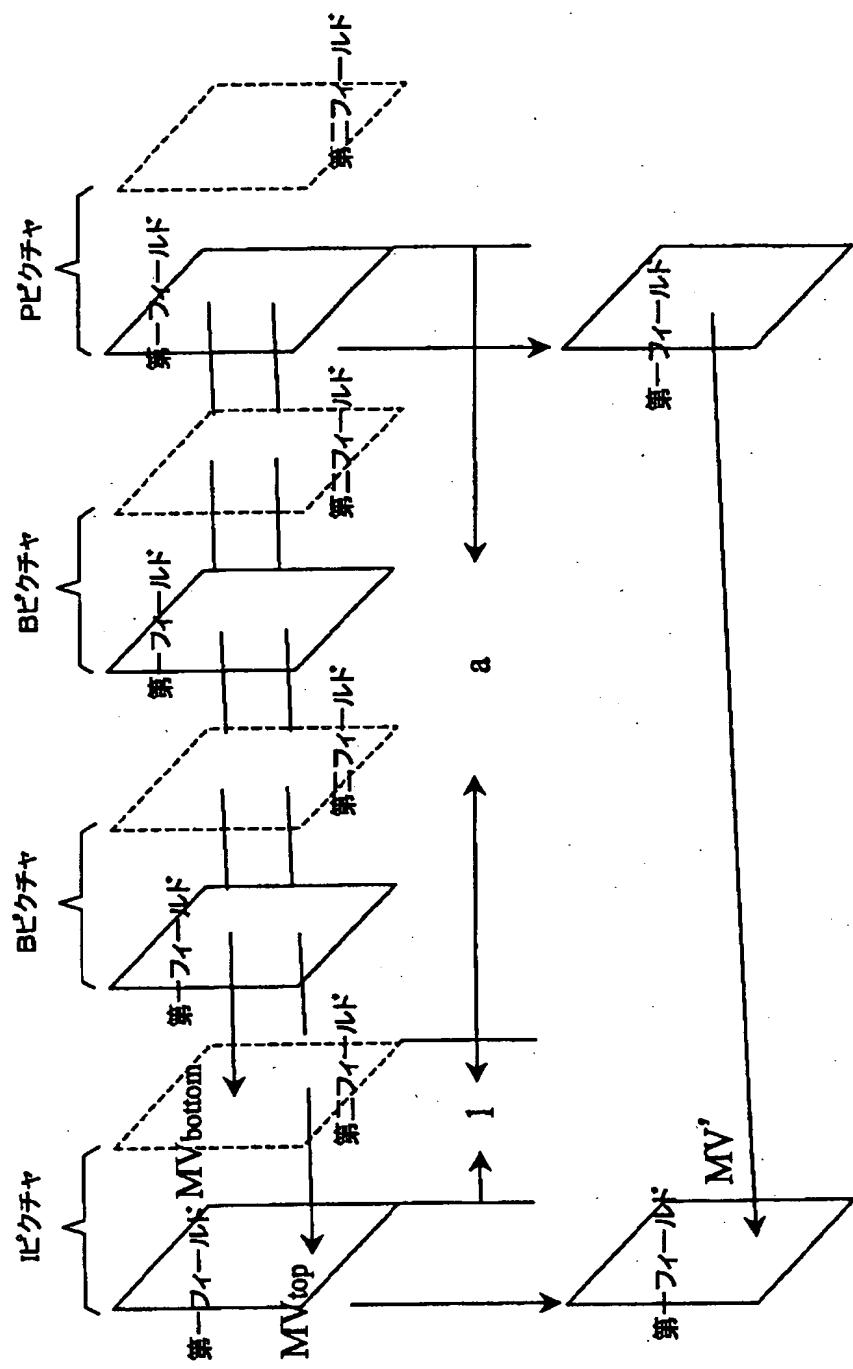
【図11】



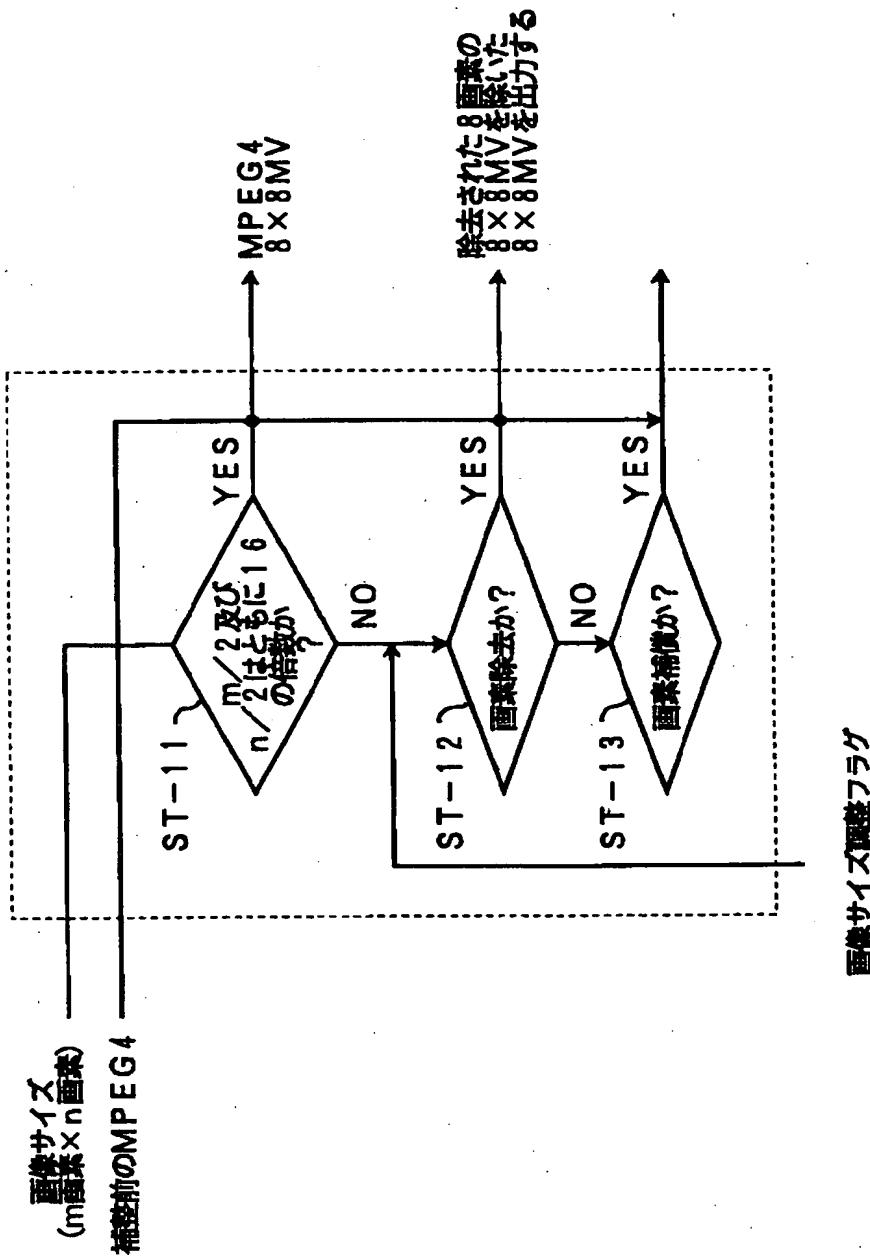
【図12】



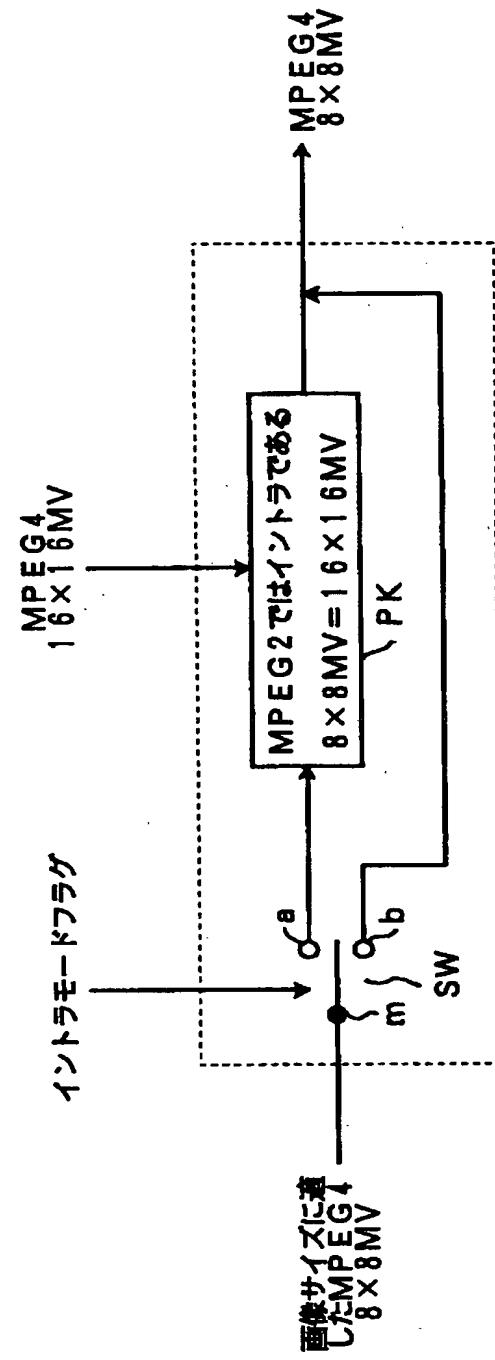
【図13】



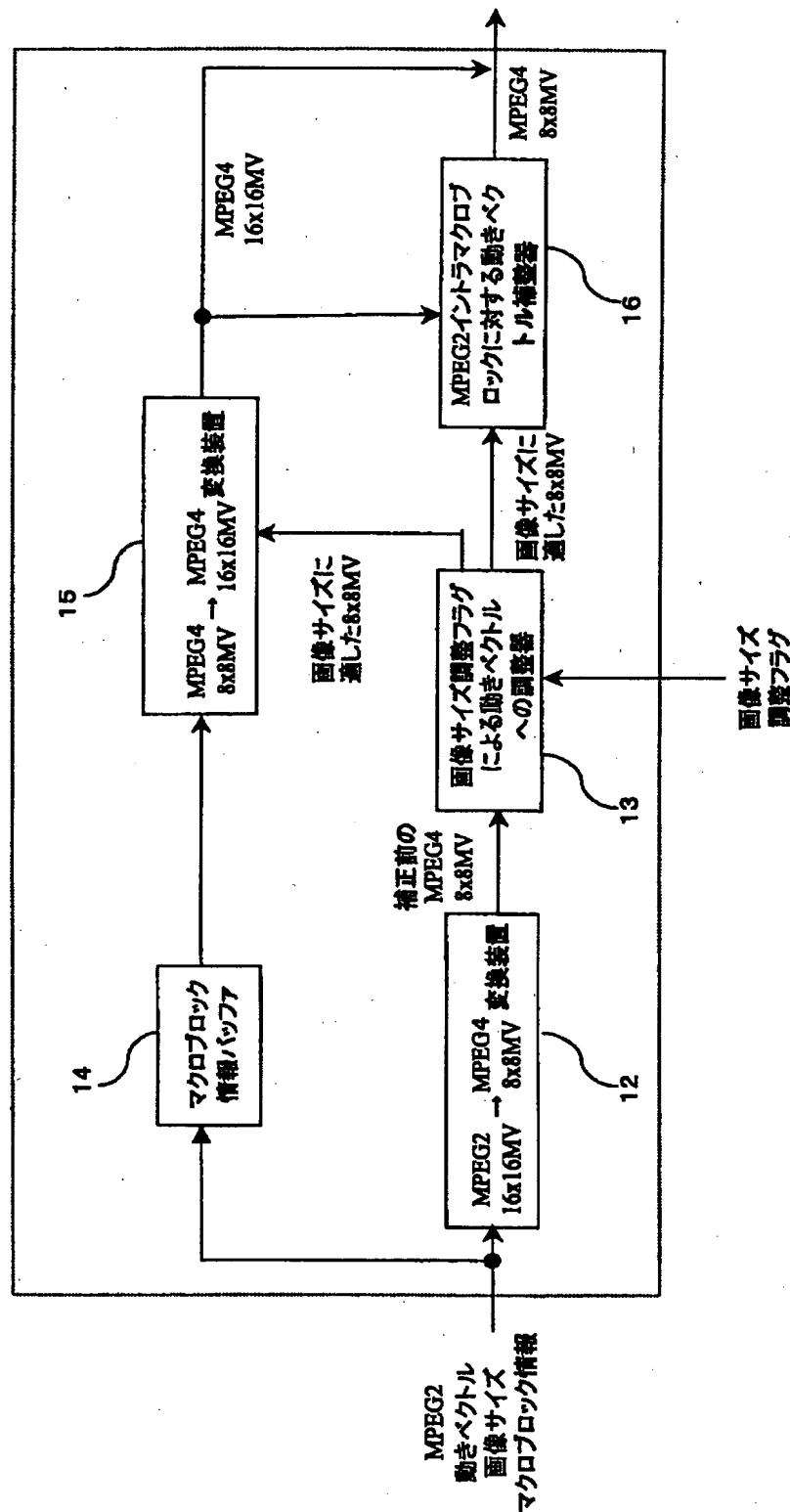
【図14】



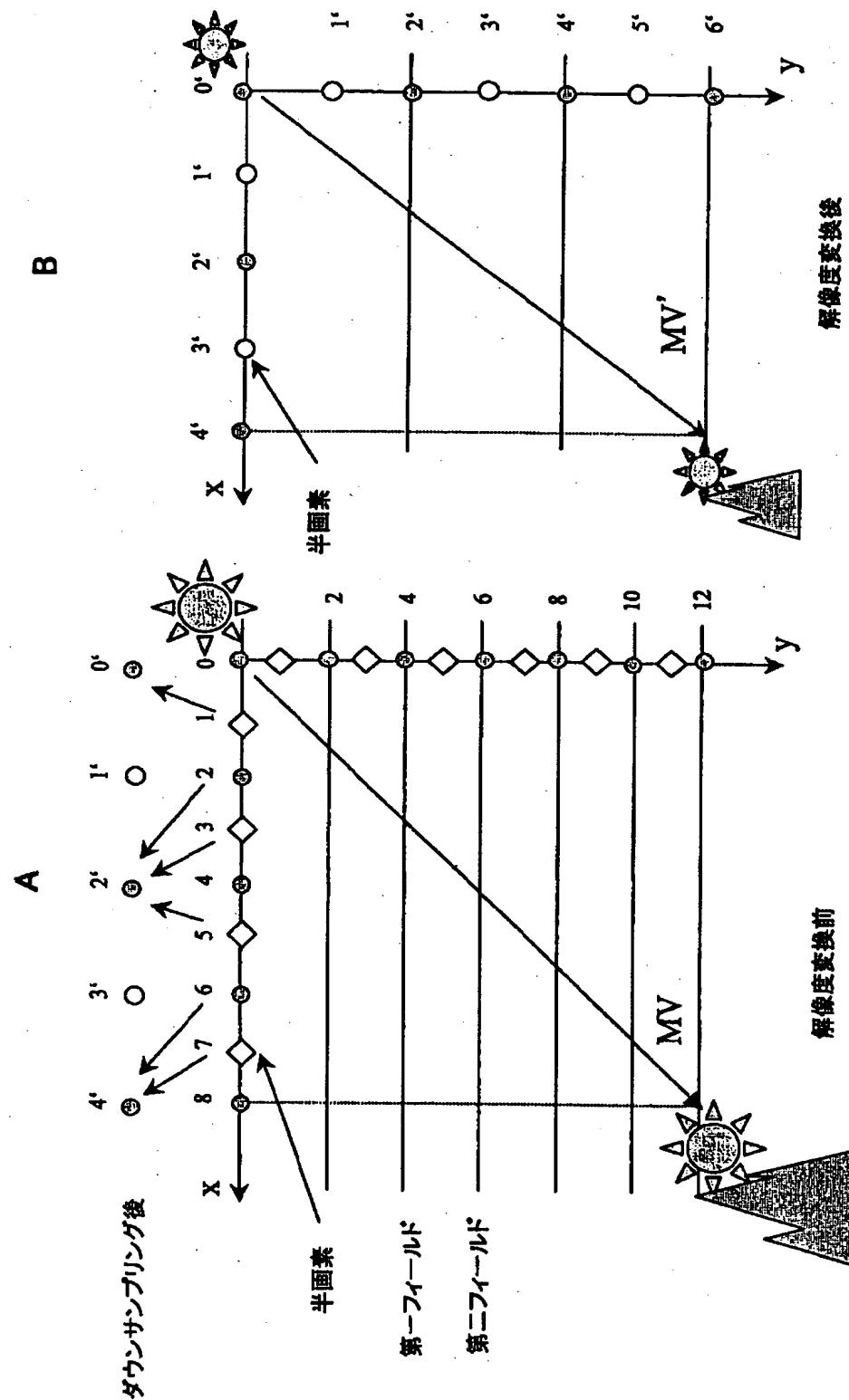
【図15】



【図16】



【図17】

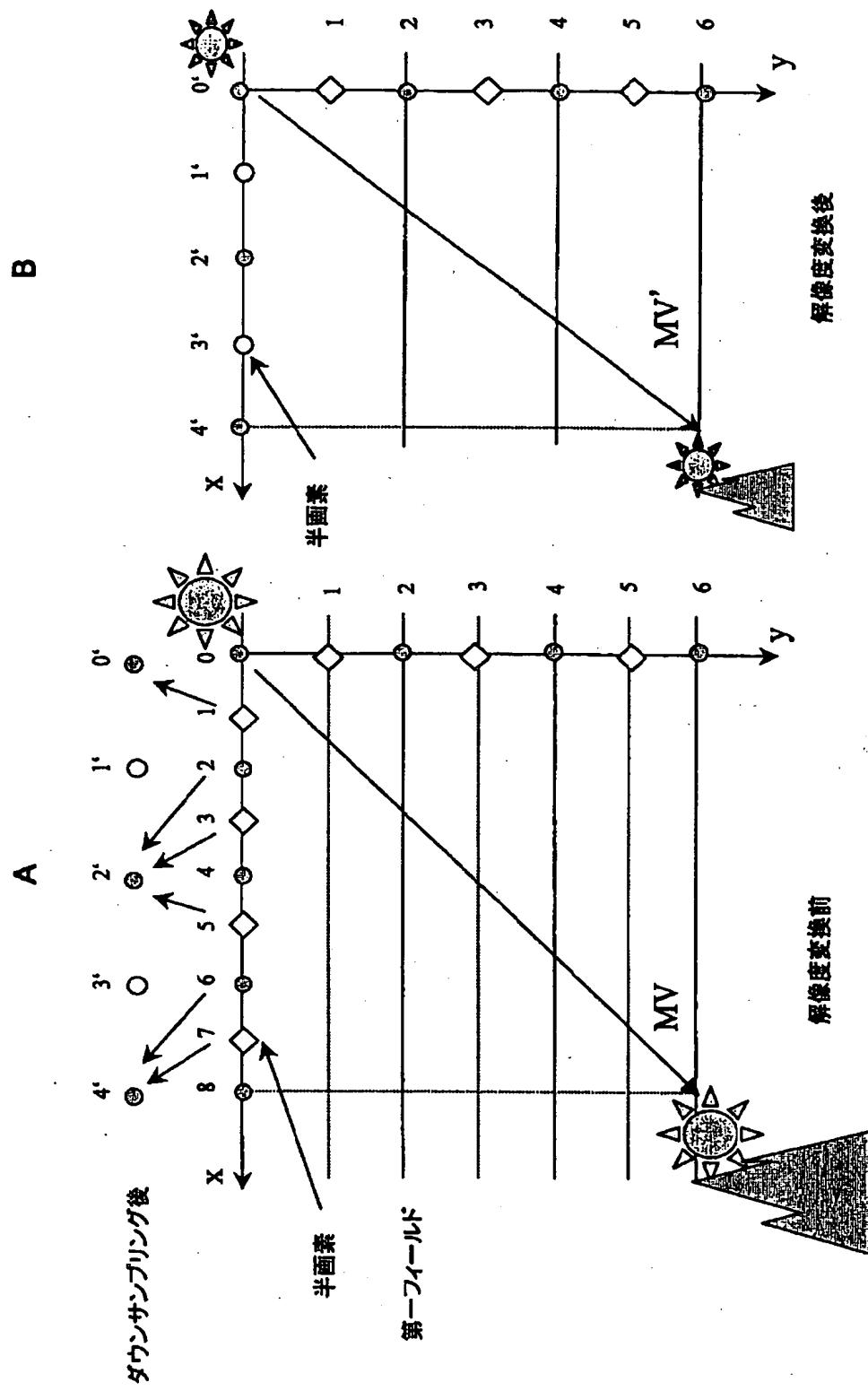


【図18】

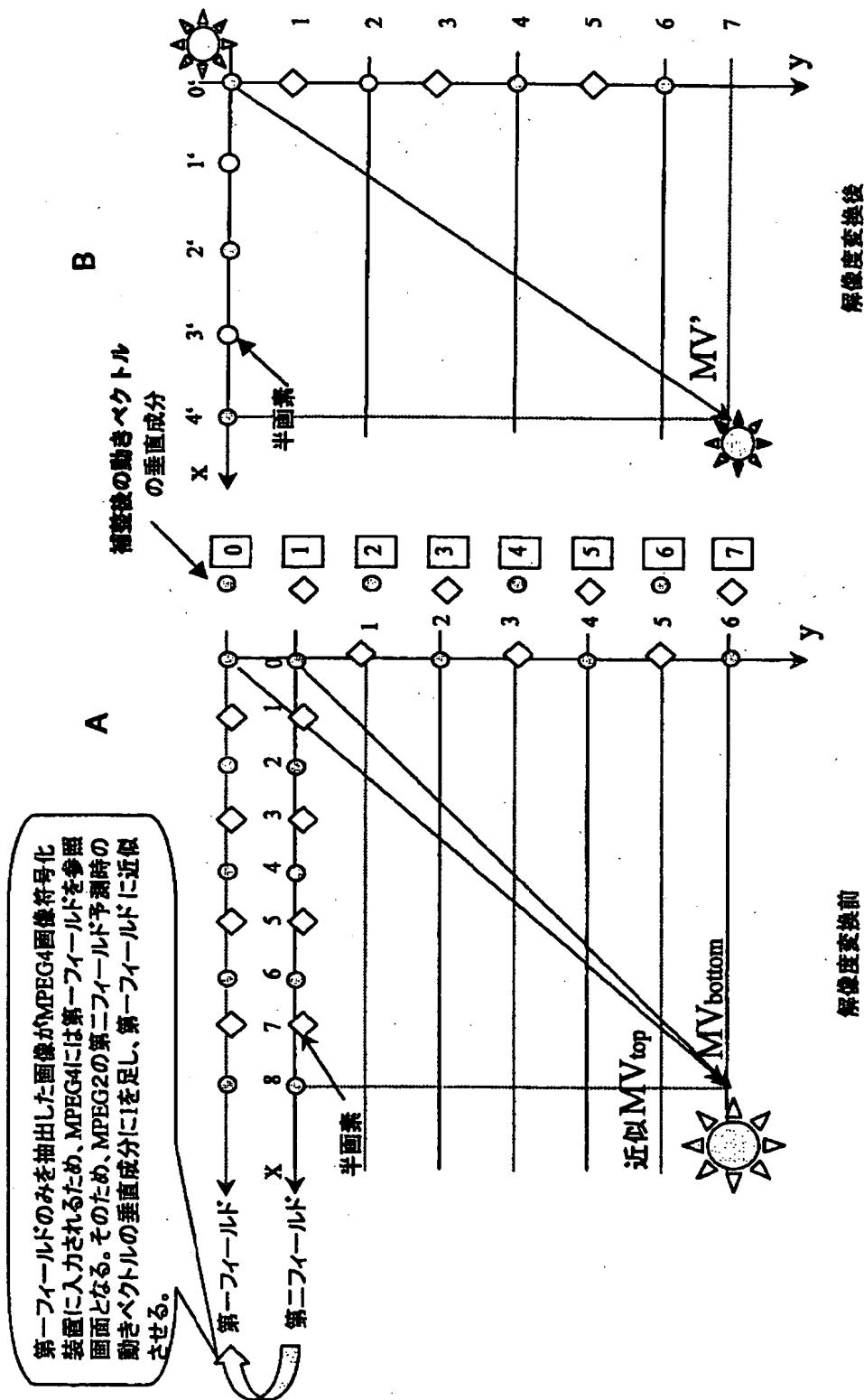
変換前の動きベクトルMV を4で割った余り	0	1	2	3
変換後の動きベクトル	[MV/2]	[MV/2]	[MV/2+1]	[MV/2]

[MV/2]はMVを2で割った整数部を表す

【図19】

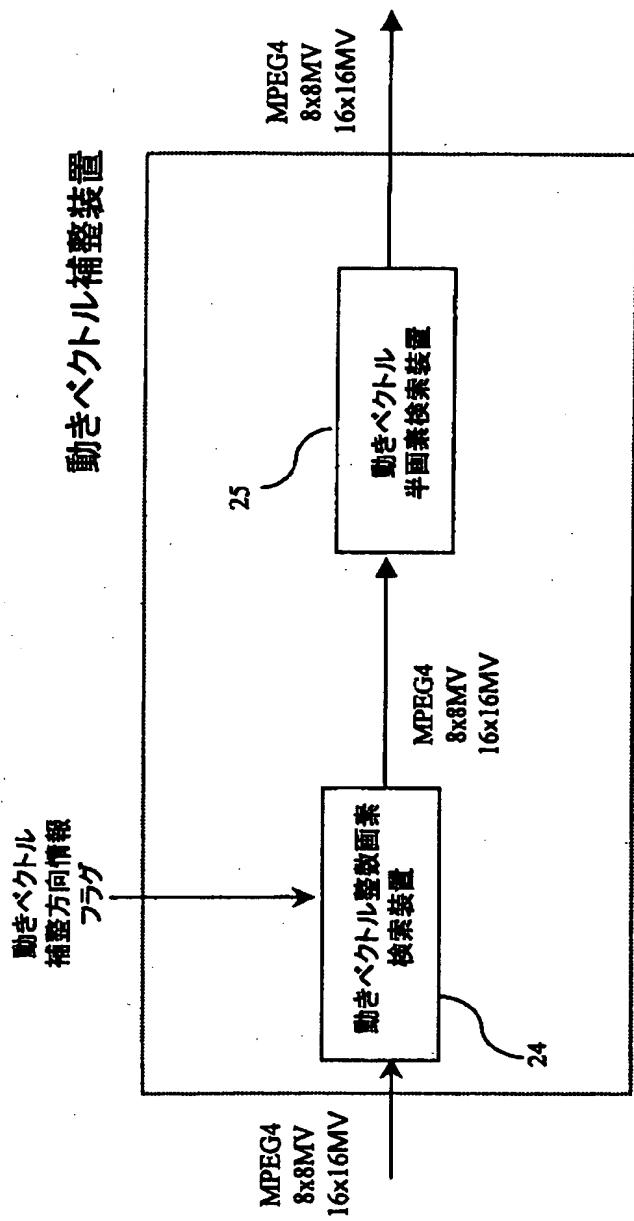


【図20】

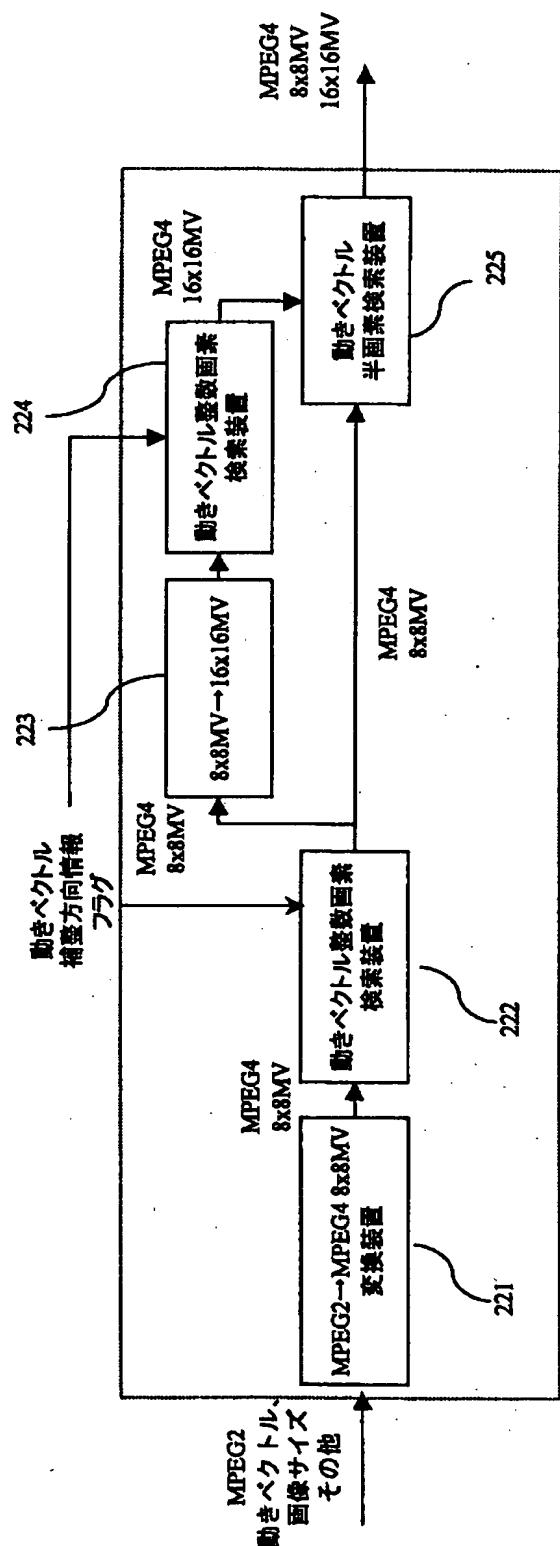


第一フィールドのみを抽出した画像がMPEG4画像符号化装置に入力されるため、MPEG4には第一フィールドを参照画面となる。そのため、MPEG2の第二フィールド予測時の動きベクトルに足し、第一フィールドに近似させる。

【図21】



【図22】



## 動きベクトル変換装置

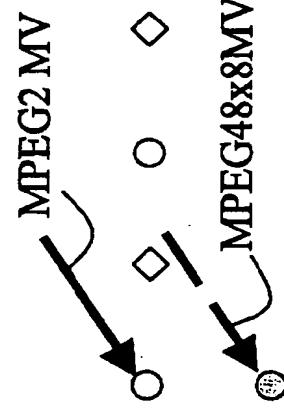
【図23】

- MPEG2 整数画素
- ◇ MPEG2 半画素
- ◎ MPEG4 整数画素

A

MPEG2 整数画素より

MPEG4整数画素に補整

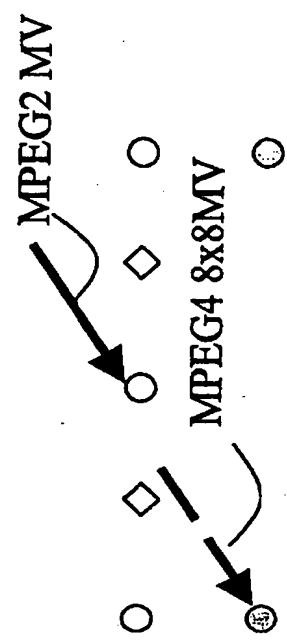


動きベクトル順方向

B

MPEG2 整数画素より

順方向のMPEG4整数画素に補整



動きベクトル順方向

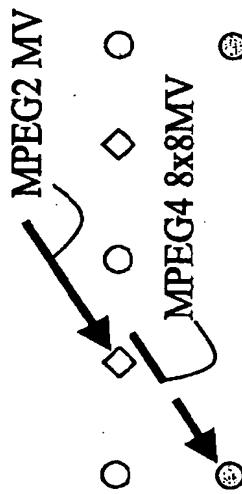
【図24】

- MPEG2 整数画素
- MPEG4 整数画素
- ◇ MPEG2 半画素
- ◇ MPEG4 8x8MV

A

MPEG2 半画素より順方向の

MPEG4 整数画素値に補整

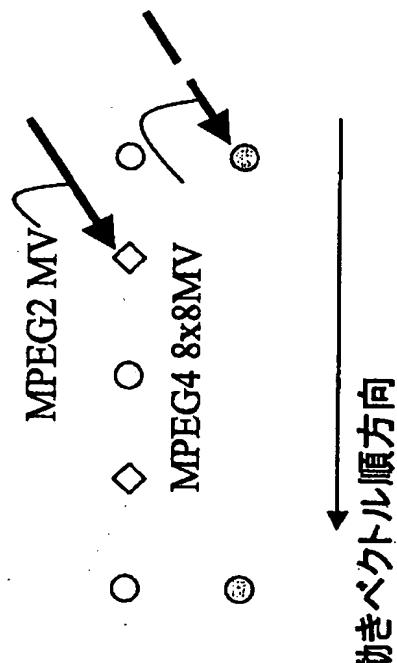


動きベクトル順方向

B

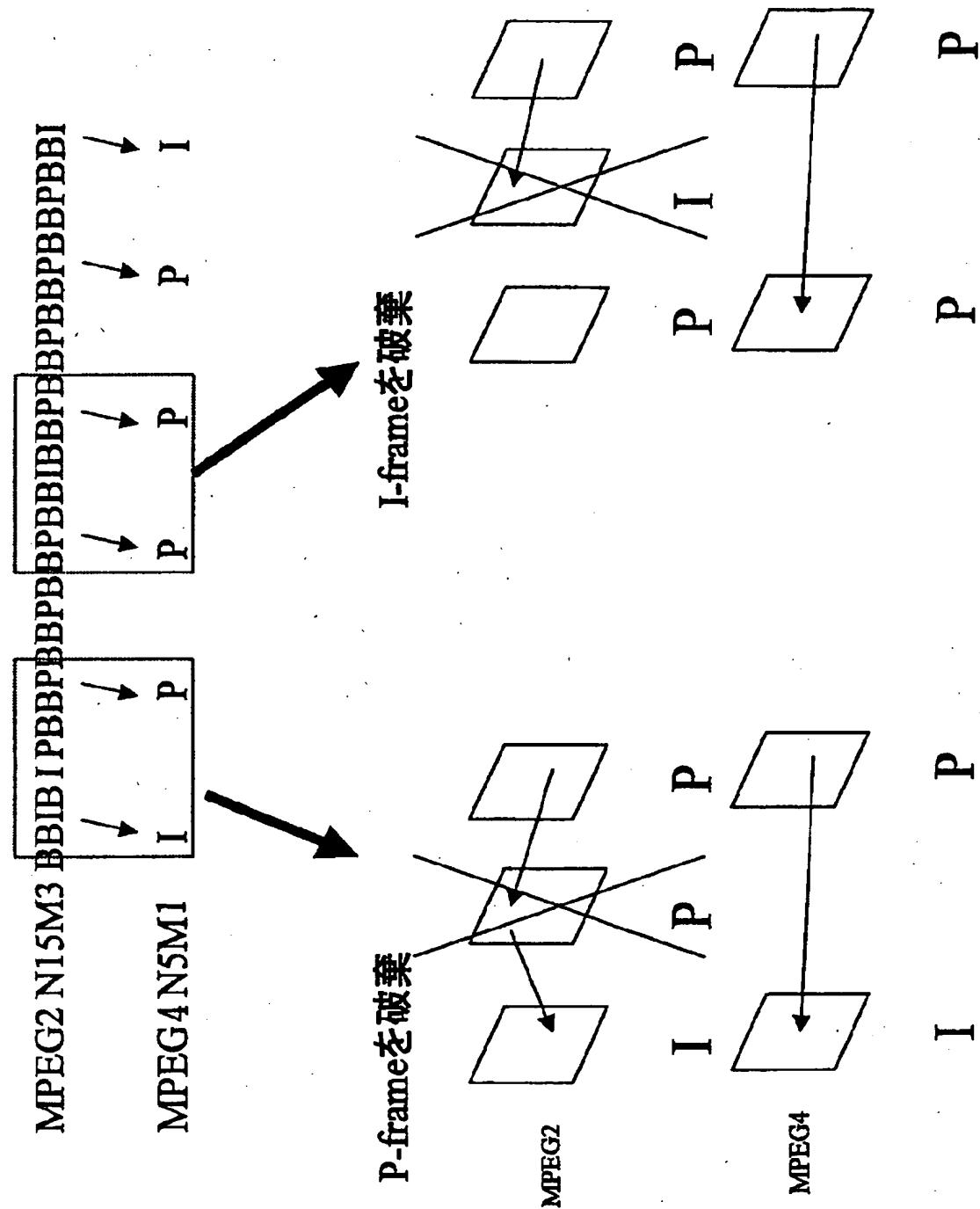
MPEG2 半画素より逆方向の

MPEG4 整数画素値に補整

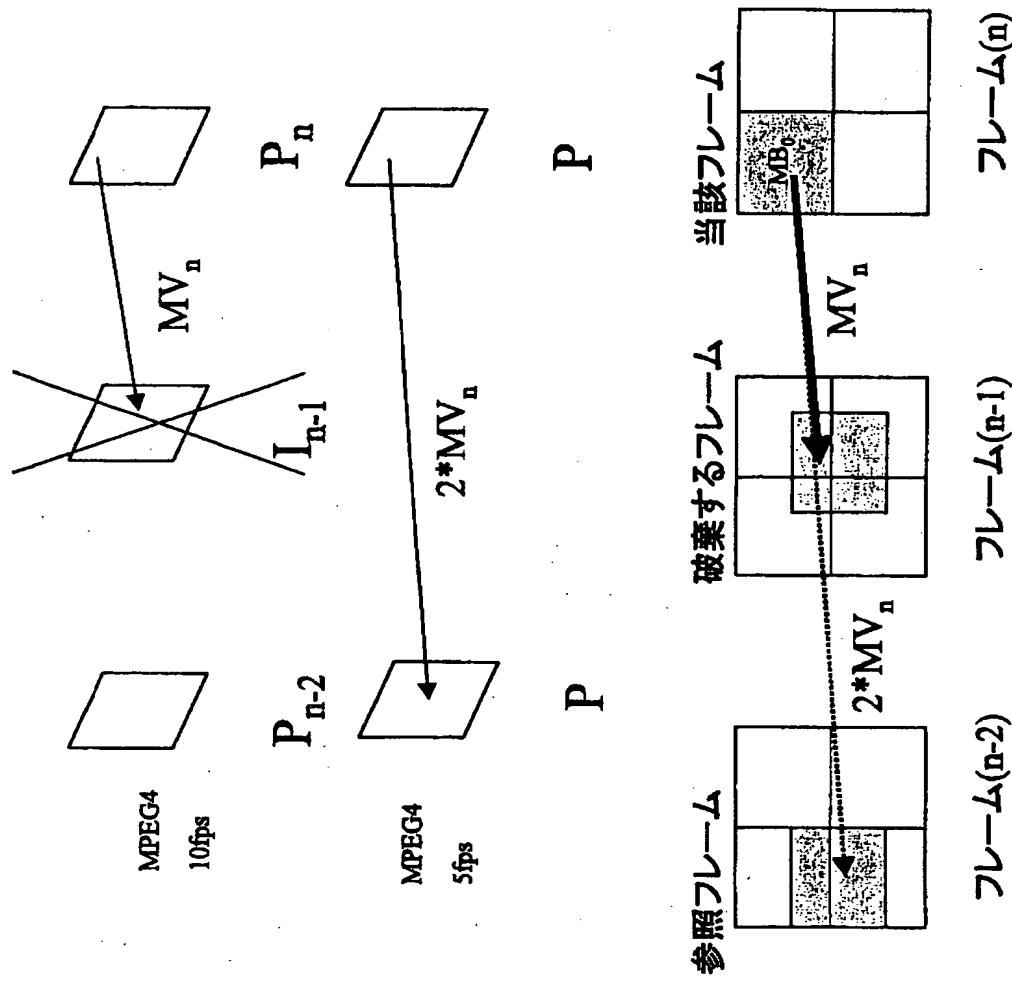


動きベクトル順方向

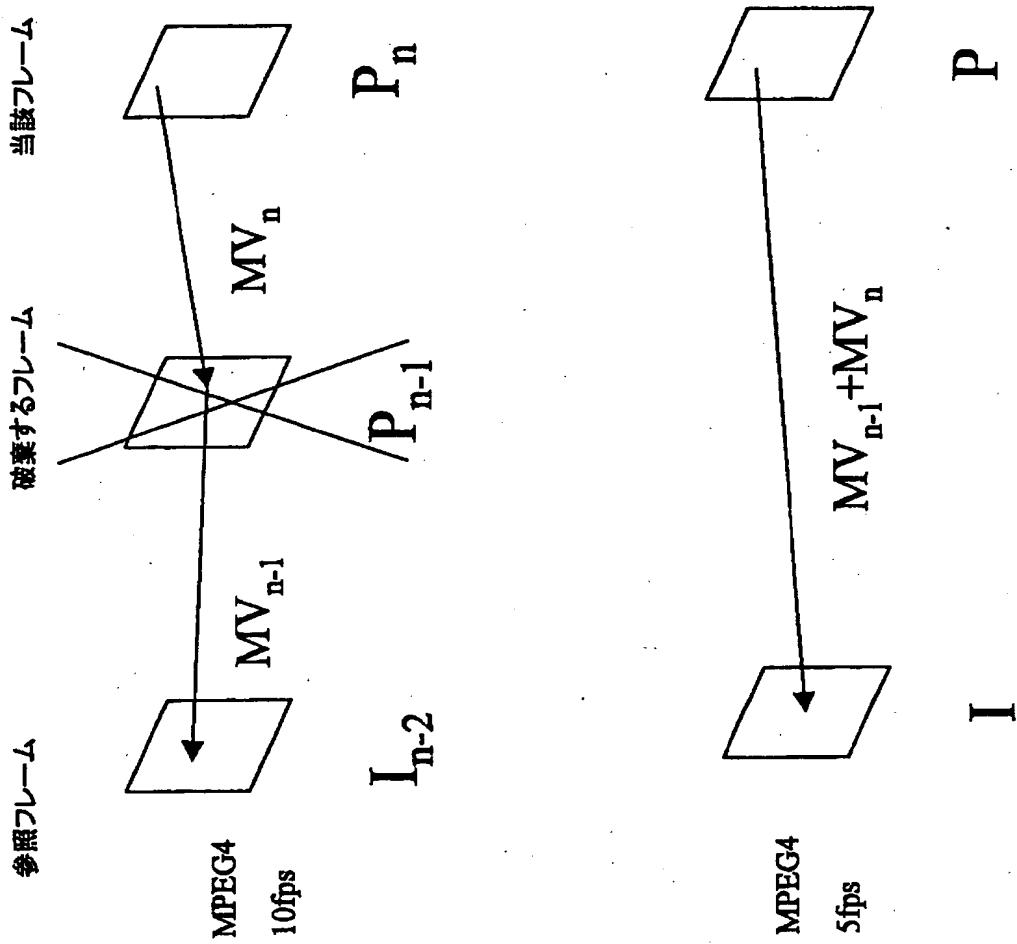
【図25】



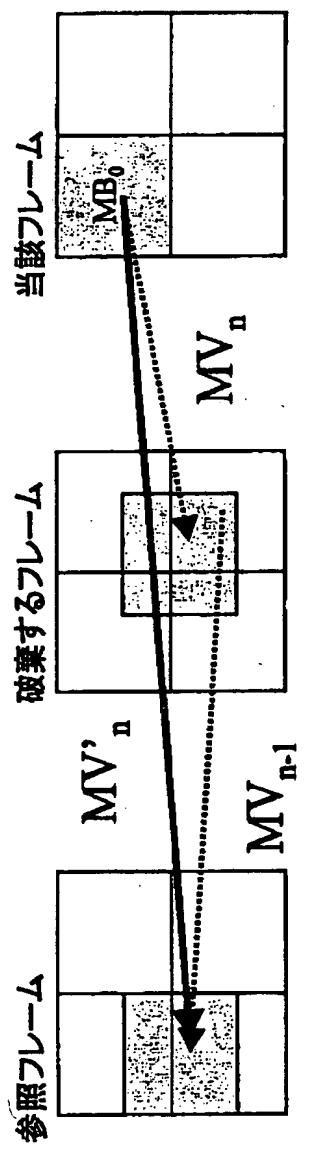
【図26】



【図27】



【図28】

Frame(n-2)  
Frame(n-1)  
Frame(n)

パラメータ  $X$  が最大となる  $MV_{n-1}$   
(ただし、 $X$  は下記のいずれか)

- MB overlapped area

- MB overlapped area / Coef bits

- MB overlapped area / Q-scale

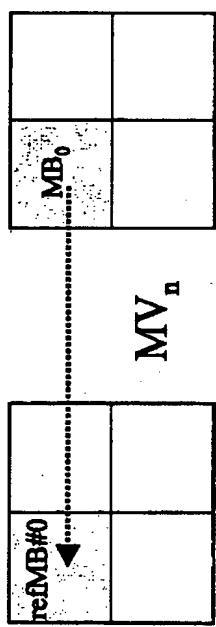
- MB overlapped area / (Coef bits × Q-scale)

$$MV'_n = MV_n + MV_{n-1}$$

【図29】

## Overlap 1 MBs

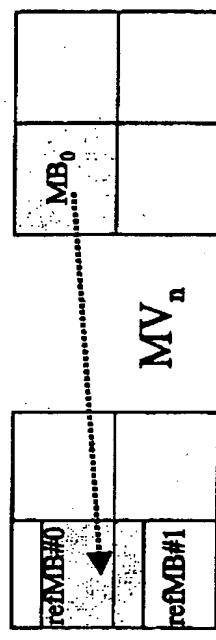
Dropped frame



Frame(n-1) Frame(n)

## Overlap 2 MBs

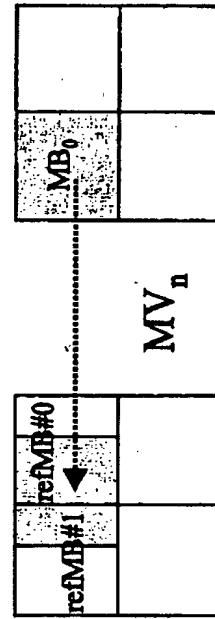
Dropped frame



Frame(n-1) Frame(n)

## Overlap 2 MBs

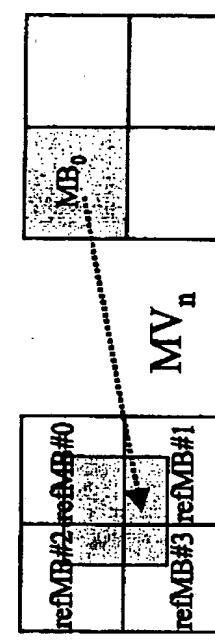
Dropped frame



Frame(n-1) Frame(n)

## Overlap 4 MBs

Dropped frame

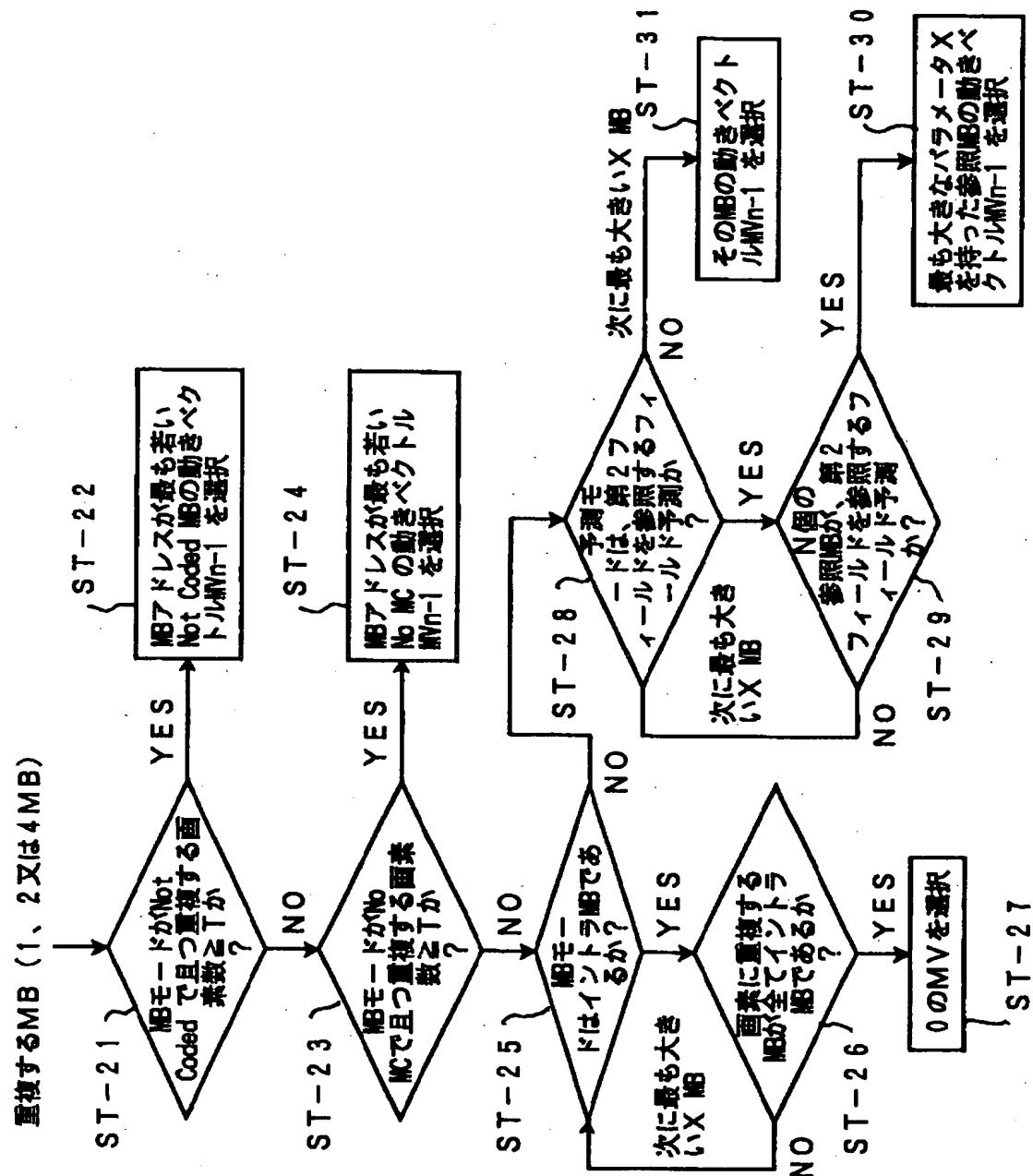


Frame(n-1) Frame(n)

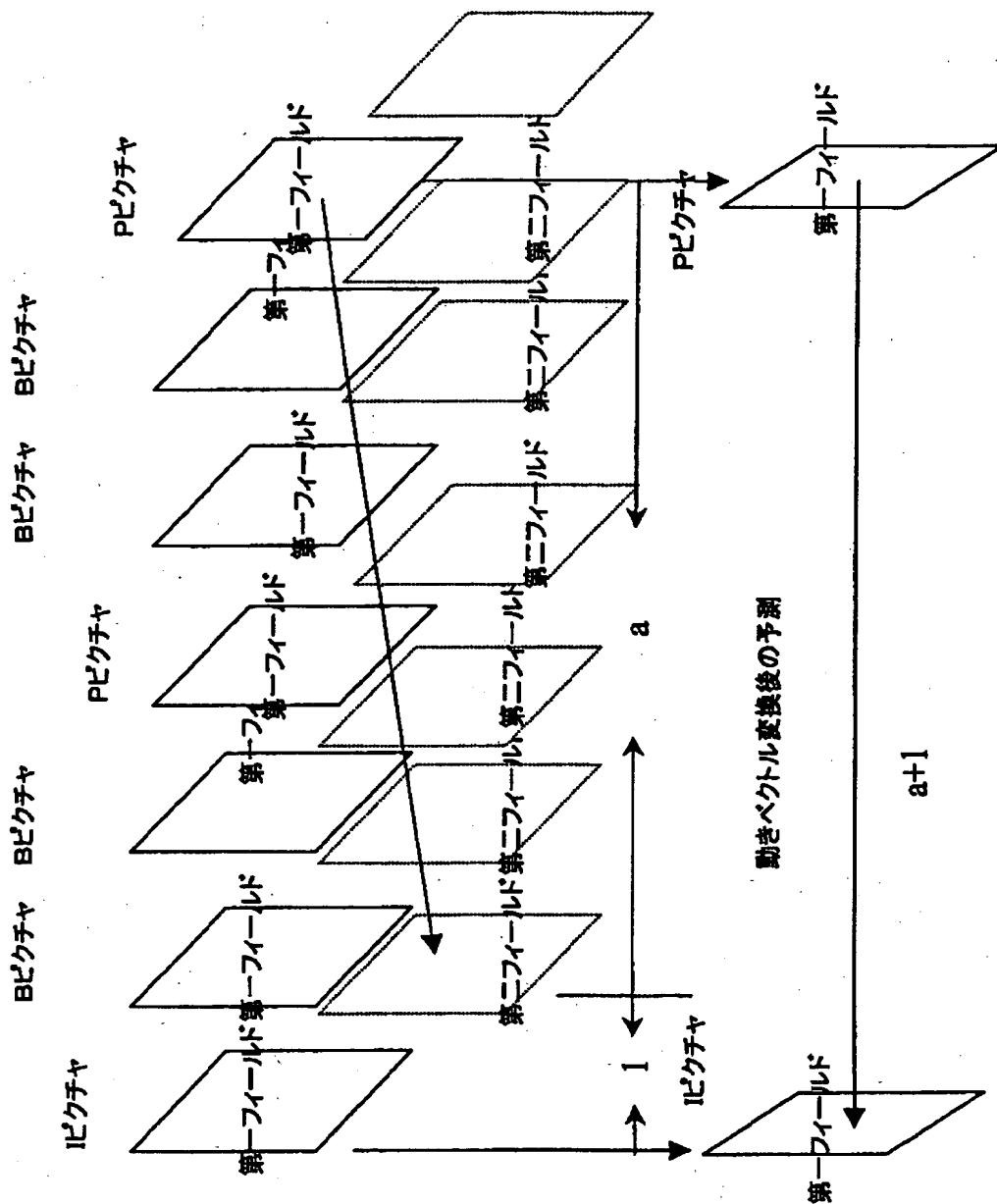
Frame(n-1) Frame(n)

Frame(n-1) Frame(n)

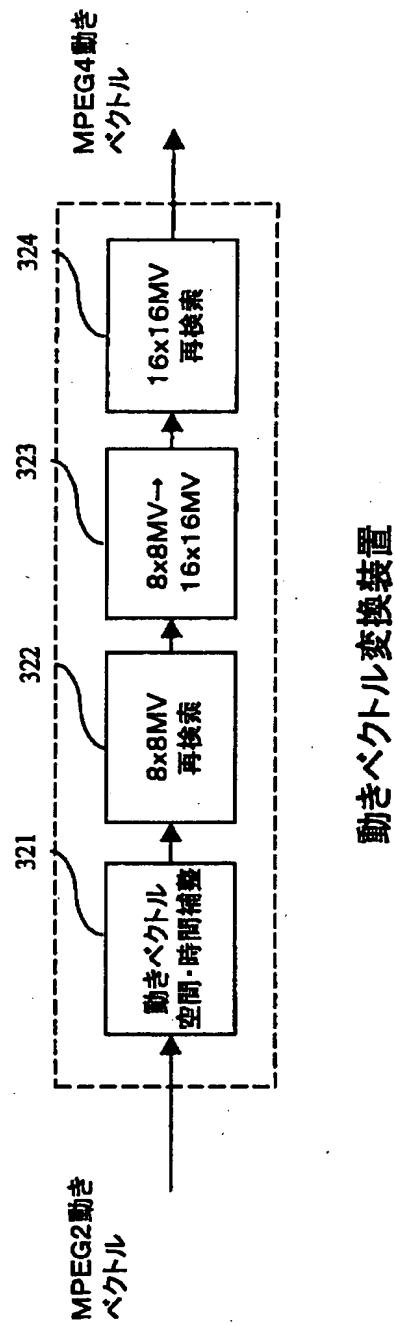
【図30】



【図31】

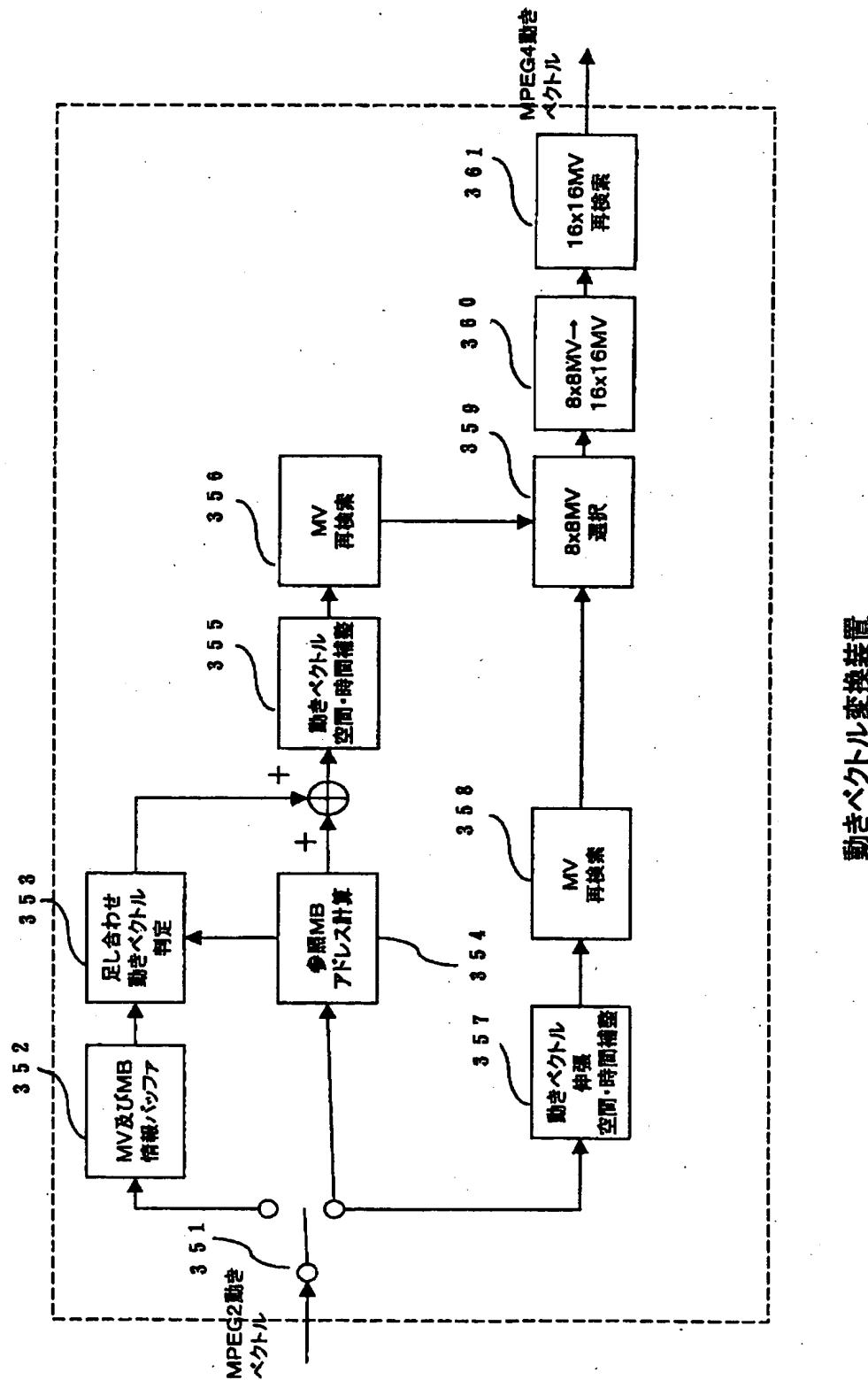


【図32】



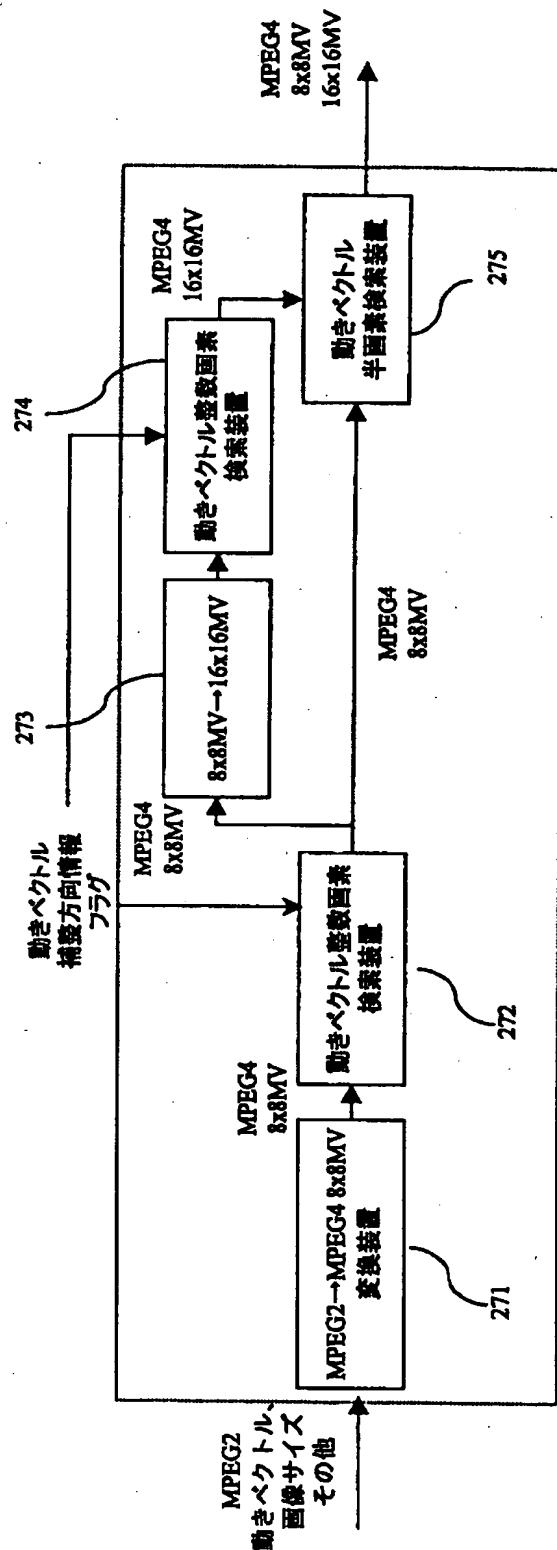
動きベクトル変換装置

【図33】



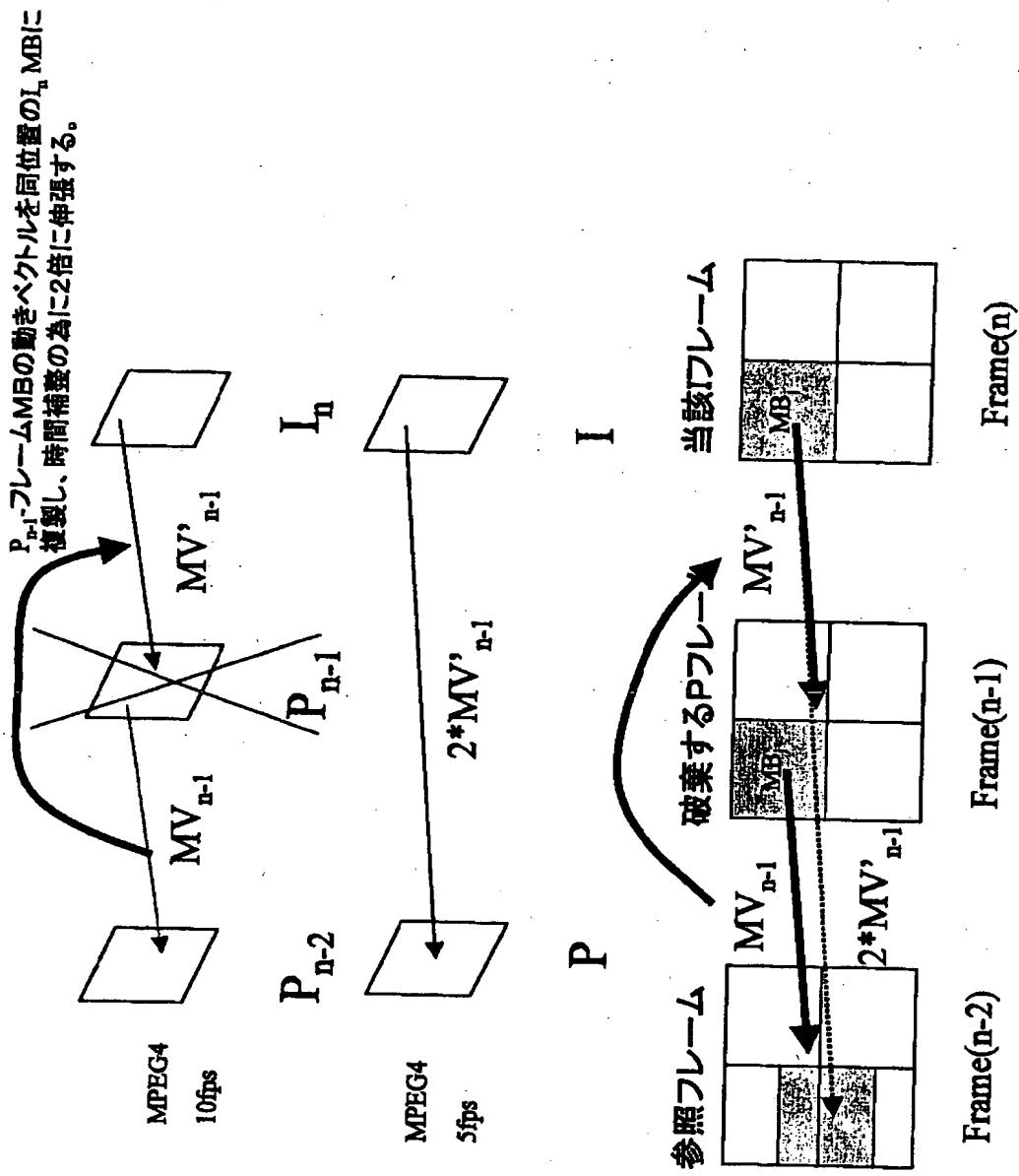
動きベクトル変換装置

【図34】



## 動きベクトル変換装置

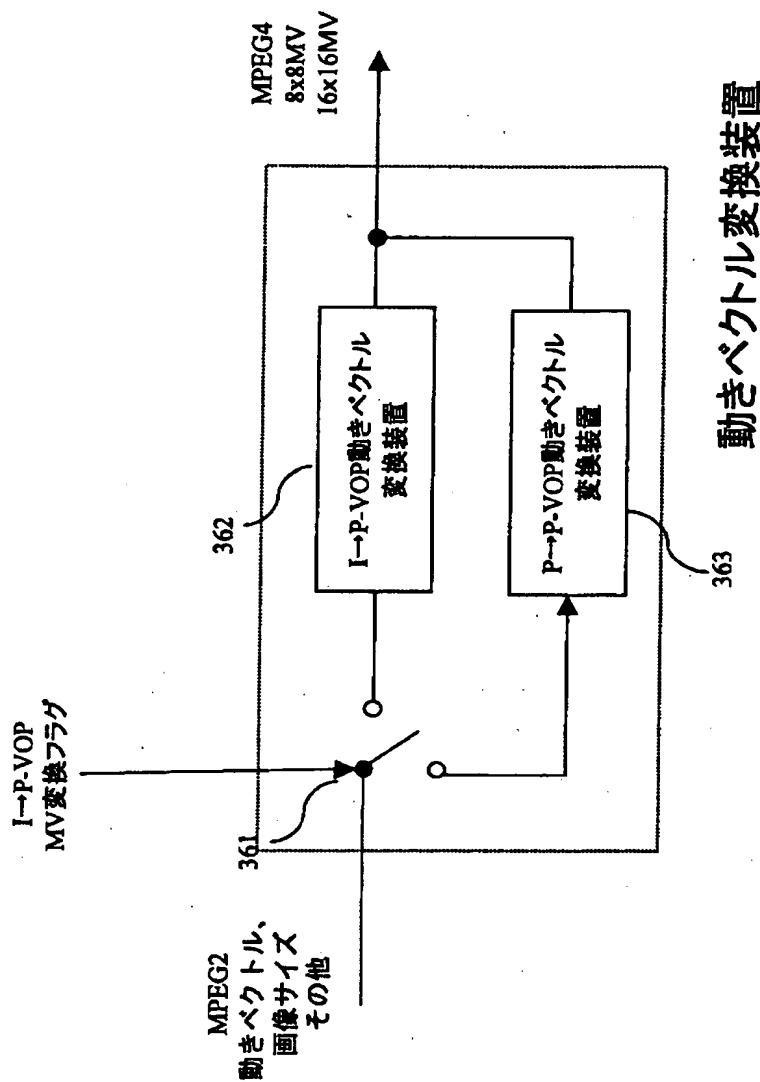
【図35】



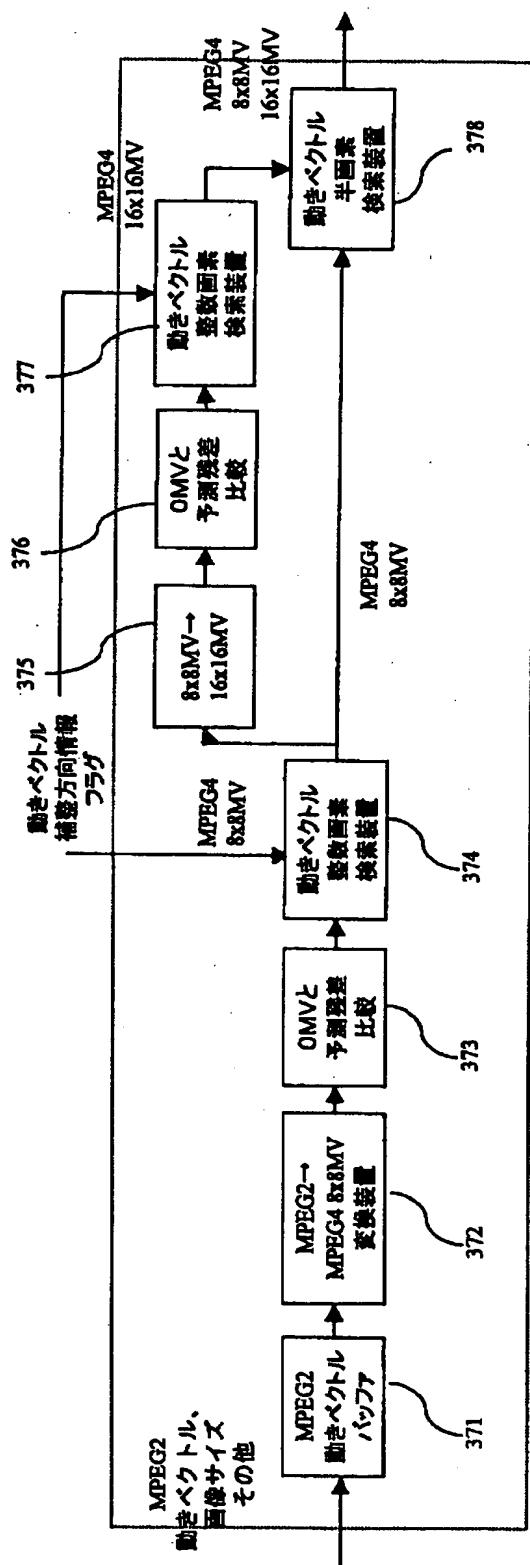
$MV_{n-1}$  を複製し、時間補整の為2倍に伸張する。

Frame(n-2) Frame(n-1) Frame(n)

【図36】



【図37】



I → P動きベクトル変換装置

【書類名】 要約書

【要約】

【課題】 画像情報変換方法におけるMPEG4の画像符号化の符号化効率を一層高くすることのできるものを得る。

【解決手段】 MPEG2のビットストリームのPフレームを一枚置きに破棄することにより、フレームレートを下げた、低ビットレートのMPEG4のビットストリームを生成する動きベクトル変換方法であって、動きベクトル補整を兼ね、入力となるMPEG2の画像圧縮情報を示すビットストリーム内のマクロブロック及びブロック毎の情報を予め格納しておき、その格納された情報を基に、破棄する直前のPフレームの動きベクトルを複製し、その複製された動きベクトルを時間方向に2倍に伸張して、MPEG2のイントラフレームより変換されたMPEG4の動きベクトルを生成する。

【選択図】 図35

出願人履歴情報

識別番号 [000002185]

1. 変更年月日 1990年 8月30日

[変更理由] 新規登録

住 所 東京都品川区北品川6丁目7番35号

氏 名 ソニー株式会社